



UNIVERSIDAD TÉCNICA DE ESMERALDAS LUIS VARGAS TORRES

FACULTAD DE INGENIERIAS

PROGRAMACION AVANZADA II



TAREA #3

DOCENTE: Ing. Johnny Quiñonez Q.

NOMBRE(S): Oscar Andrés Acurio Villacrés, Roberto Daniel García Cedeño, Jennifer Fiorella Castillo Cabeza

FECHA: 21/10/2020

NIVEL: Octavo

PARALELO: “B”

TEMA GENERAL

APLICACIÓN EN N-CAPAS

ÍNDICE

1. INTRODUCCIÓN.....	3
2. OBJETIVO	3
3. DESARROLLO	4
3.1. INSTALACIÓN DE SQL SERVER 2019.....	4
3.2. COPIAR Y PEGAR BASE DE DATOS EN SQL SERVER 2019.....	12
3.3. CREAR Y CONFIGURAR UN LOGIN EN SQL SERVER 2019	15
3.4. INSTALACIÓN DE VISUAL STUDIO 2019.....	17
3.5. CREACIÓN DE UN PROYECTO EN VISUAL STUDIO 2019	22
3.6. CREACIÓN DE LAS CAPAS	25
3.7. CONEXIÓN DE LA CAPA DE DATOS CON LA BASE DE DATOS ...	28
3.8. REFERENCIAR LA CAPA DE DATOS CON LA CAPA ENTIDADES	33
3.9. CREACIÓN DE LOS REPOSITORIOS PARA LA MANIPULACIÓN DE LOS ELEMENTOS DE LA BD	38
3.10. MÉTODOS CRUD DE LA CLASE “GAC_HOSPITALES_10_REPOSITORY.CS”.....	39

3.11.	CREACIÓN DEL MÉTODO CRUD DE LA CLASE “GAC_PACIENTES_10_REPOSITORY.CS”	40
3.12.	CREACIÓN DE CLASES Y MÉTODOS EN LA CAPA “GAC_LOGICA_NEGOCIO_10”	43
3.13.	PROGRAMACIÓN DE LA CAPA DE PRESENTACIÓN.....	47
3.14.	CODIFICAR EL BOTÓN GUARDAR	56
3.15.	DISEÑO CON BOOTSTRAP	58
3.16.	CAMBIAR EL MODELO DEL MENÚ.....	62
3.17.	CAMBIAR EL MODELO DE NEWHOSPITALES	63
3.18.	CREACIÓN DEL DISEÑO DE LA PAGINA BUSCAR HOSPITAL .	66
3.19.	CODIFICACIÓN DEL BOTÓN BUSCAR	70
3.20.	CODIFICACIÓN DEL BOTÓN ELIMINAR	72
3.21.	PROGRAMAR BOTÓN NUEVO Y SALIR DE LA PÁGINA SERCHHOSPITAL.....	73
3.22.	CODIFICACIÓN DEL DISEÑO ACTUALIZAR HOSPITAL.....	74

1. INTRODUCCIÓN

Microsoft SQL Server 2019 es un sistema de gestión de base de datos relacional, desarrollado por la empresa Microsoft. El lenguaje de desarrollo utilizado es **Transact-SQL**, una implementación del estándar **ANSI** del lenguaje **SQL**, utilizado para manipular y recuperar datos, crear tablas y definir relaciones entre ellas.

Puede ser configurado para utilizar varias instancias en el mismo **servidor físico**, la primera instalación lleva generalmente el nombre del **servidor**, y las siguientes - **nombres** específicos (con un guion invertido entre el nombre del **servidor** y el **nombre** de la instalación).

Microsoft Visual Studio 2019 es un entorno de desarrollo integrado, es compatible con múltiples lenguajes de programación, tales como **C++, C#, Visual Basic .NET, F#, Java, Python, Ruby y PHP**, al igual que entornos de desarrollo web, como **ASP.NET MVC, Django, etc.**, a lo cual hay que sumarle las nuevas capacidades en línea bajo **Windows Azure** en forma del editor **Monaco**.

Visual Studio permite a los desarrolladores **crear sitios** y **aplicaciones web**, así como servicios web en cualquier entorno compatible con la **plataforma .NET**. Así, se pueden **crear aplicaciones** que se comuniquen entre estaciones de trabajo, **páginas web, dispositivos móviles, dispositivos embebidos y videoconsolas, entre otros**.

N Capas es una arquitectura **cliente-servidor** en el que el objetivo primordial es la separación de la **lógica de negocios** de la **lógica de diseño**; un ejemplo básico de esto consiste en separar la capa de datos de la capa de presentación al usuario.

La **ventaja principal** de este estilo es que el desarrollo se puede llevar a cabo en varios niveles y, en caso de que sobrevenga algún cambio, solo se ataca al nivel requerido sin tener que revisar entre código mezclado, **Capa de presentación, Capa de Negocio, Capa de datos**.

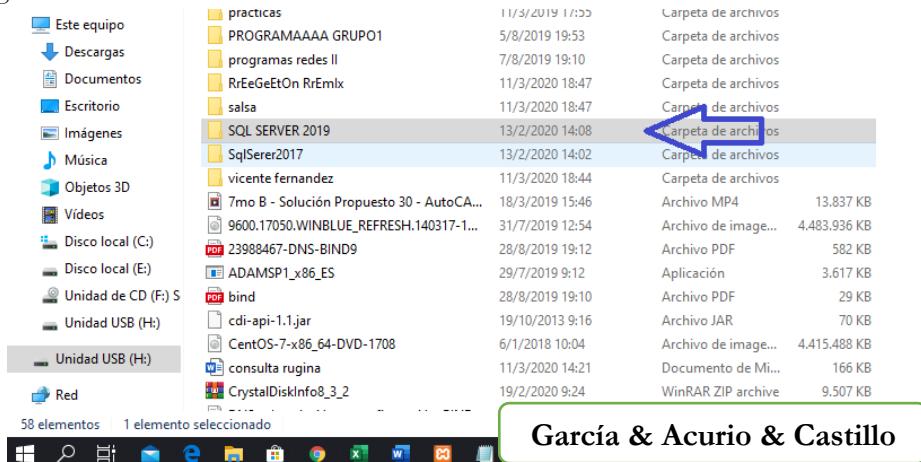
2. OBJETIVO

- Elaborar un Manual de una aplicación con **Entity Framework** y **Asp.net** usando **SQL Server 2019, Visual Studio 2019, Bootstrap** basado en programación **N-Capas**

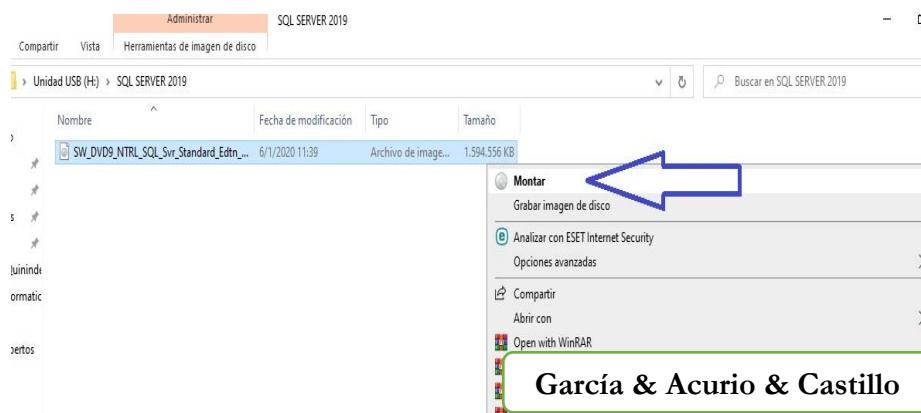
3. DESARROLLO

3.1. INSTALACIÓN DE SQL SERVER 2019

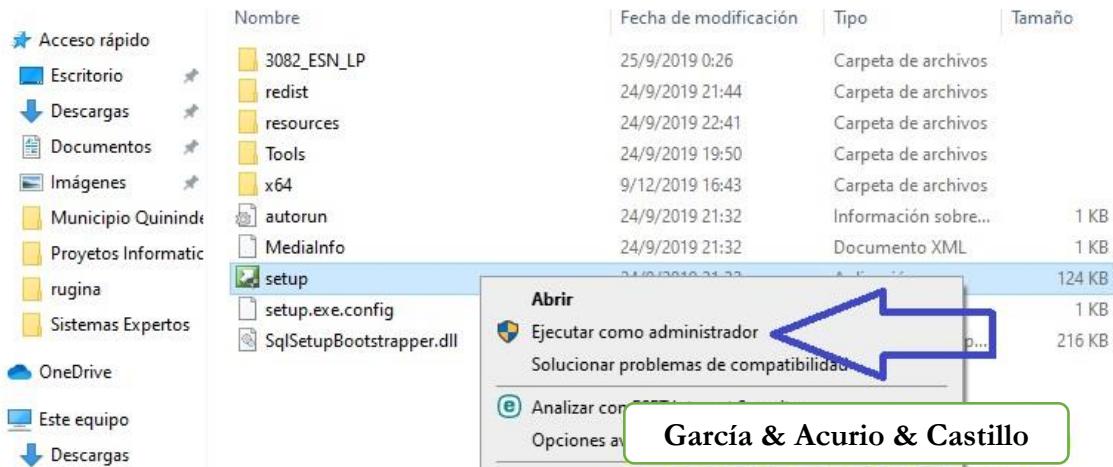
1. Nos dirigimos al directorio de **SQL SERVER 2019** le damos dos clics izquierdos para ingresar.



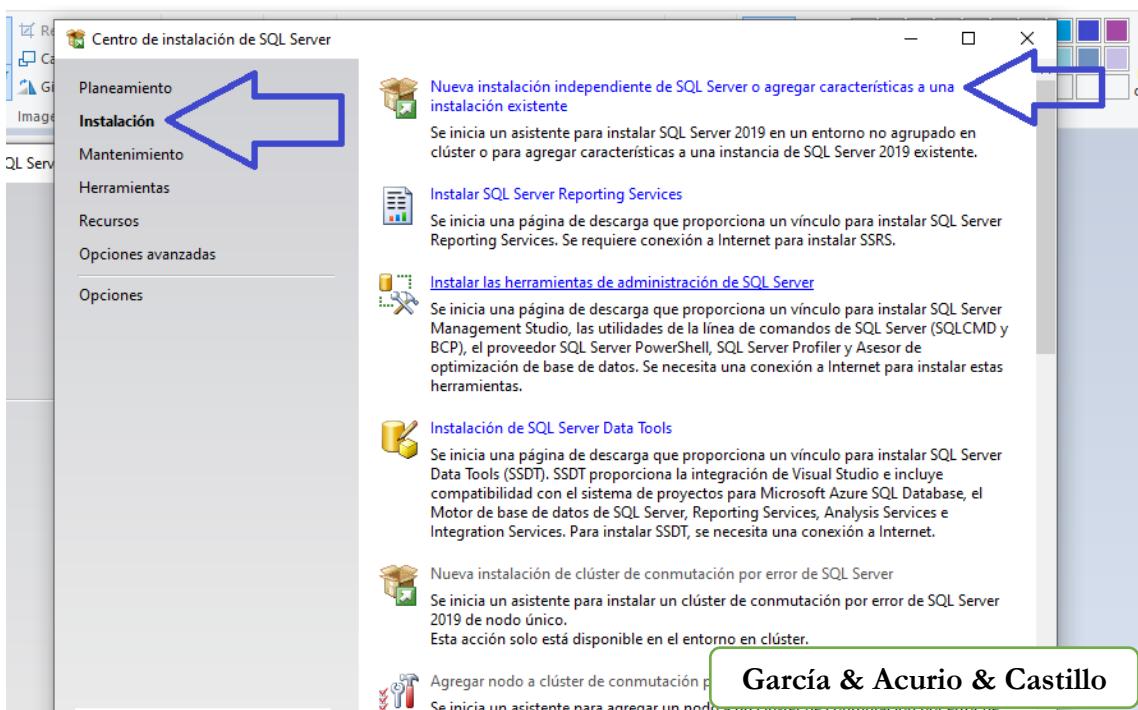
2. Una vez adentro de la carpeta de **SQL SERVER 2019** nos abre un archivo iso el cual con un programa el ultraiso podemos ejecutarlo dando clic derecho en montar.



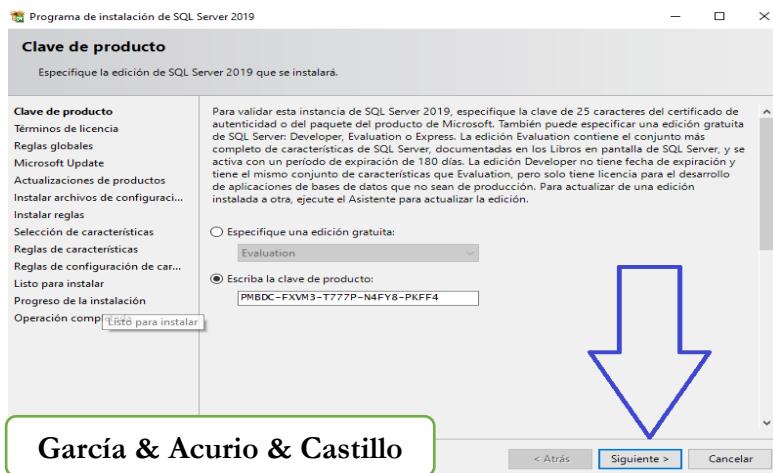
3. Luego nos aparece una ventana donde encontraremos el Setup de instalación le damos clic derecho en la opción **ejecutar como administrador** y damos clic en aceptar.



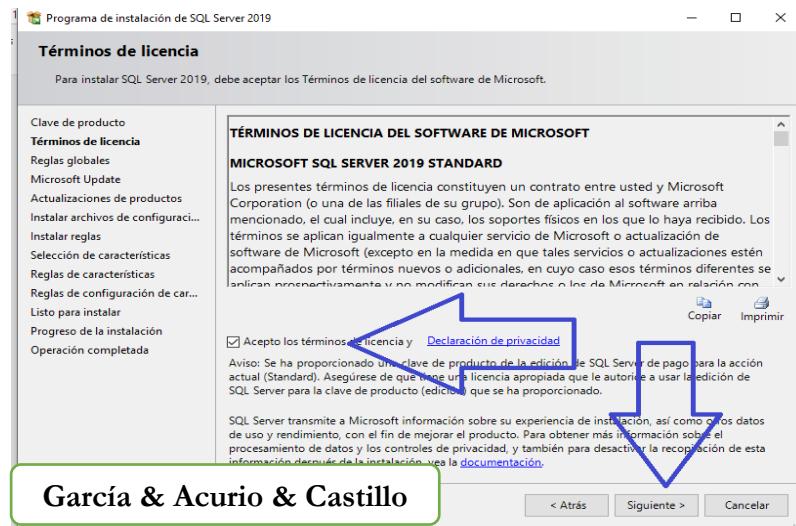
4. Luego nos abre el centro de instalación de **SQL Server**, donde nos dirigimos a la opción instalar y prosigue a nueva instalación independiente **SQL Server**.



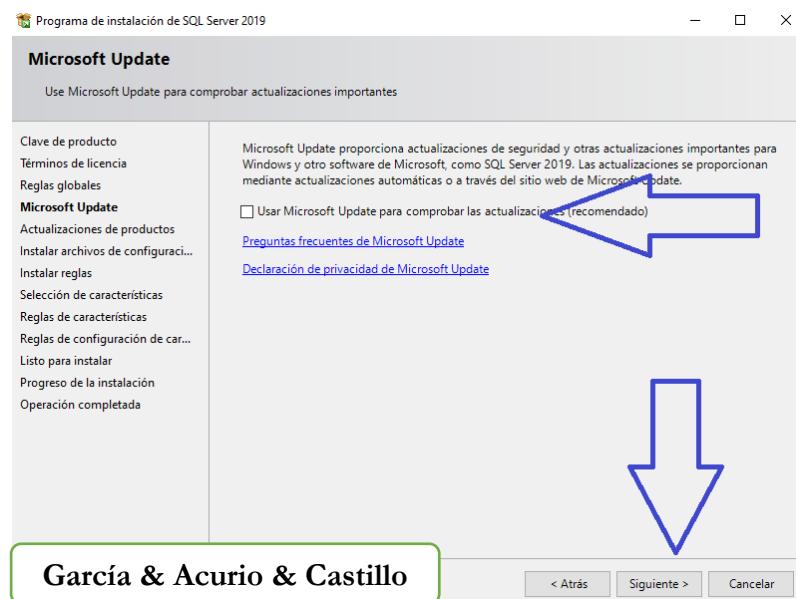
5. Se nos abre otra ventana donde ya nos da la clave del producto y le damos siguiente



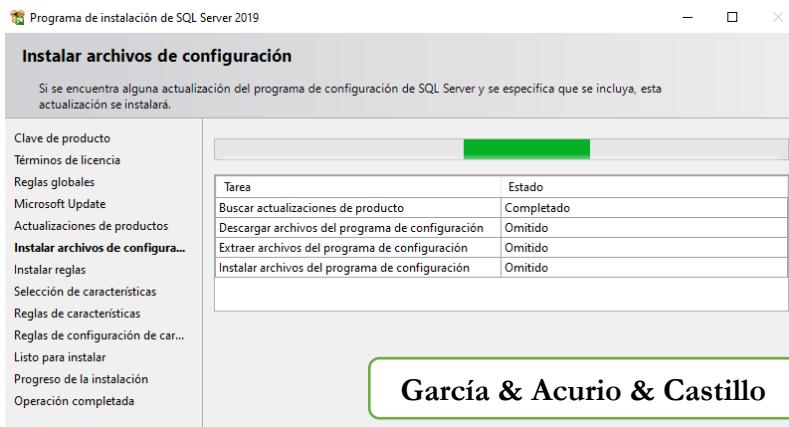
6. Luego aceptamos los créditos y le damos siguiente.



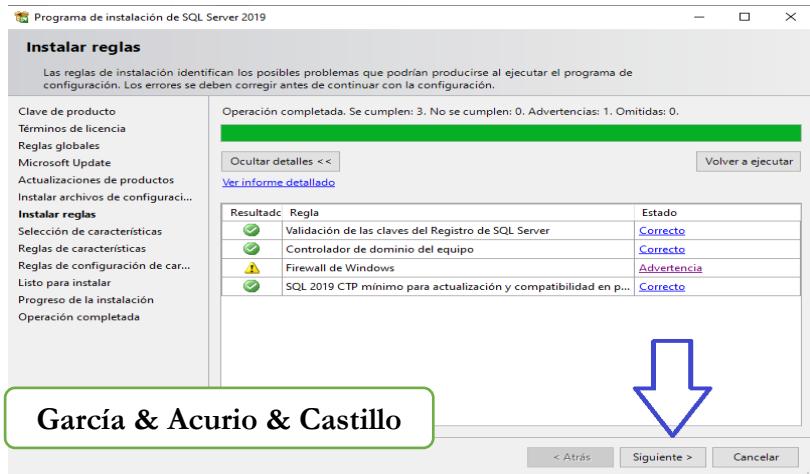
7. luego si queremos actualizamos si no le damos en siguiente



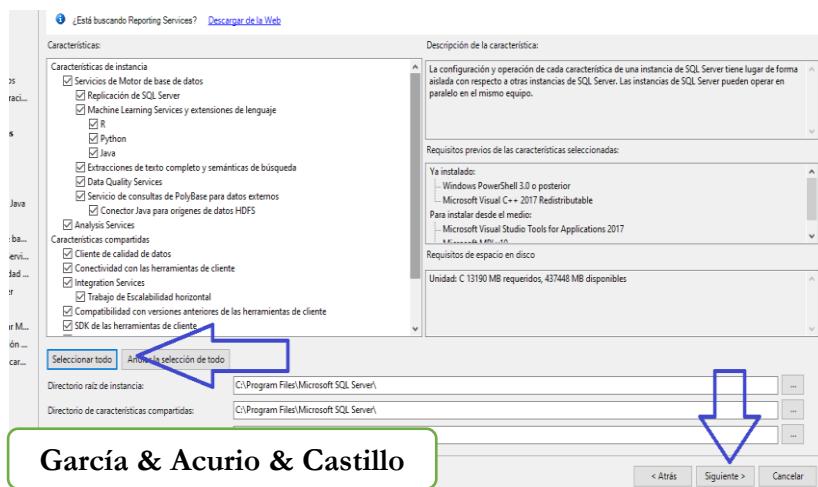
8. luego esperamos que instale archivos de configuración.



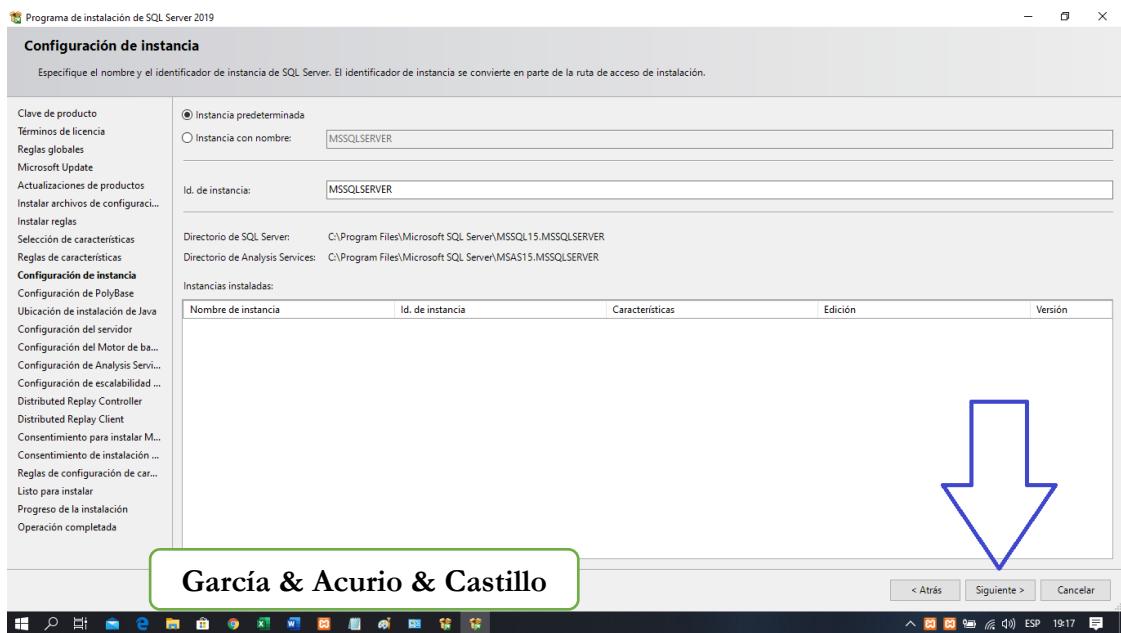
9. luego esperamos que siga instalando las configuraciones y le damos siguiente



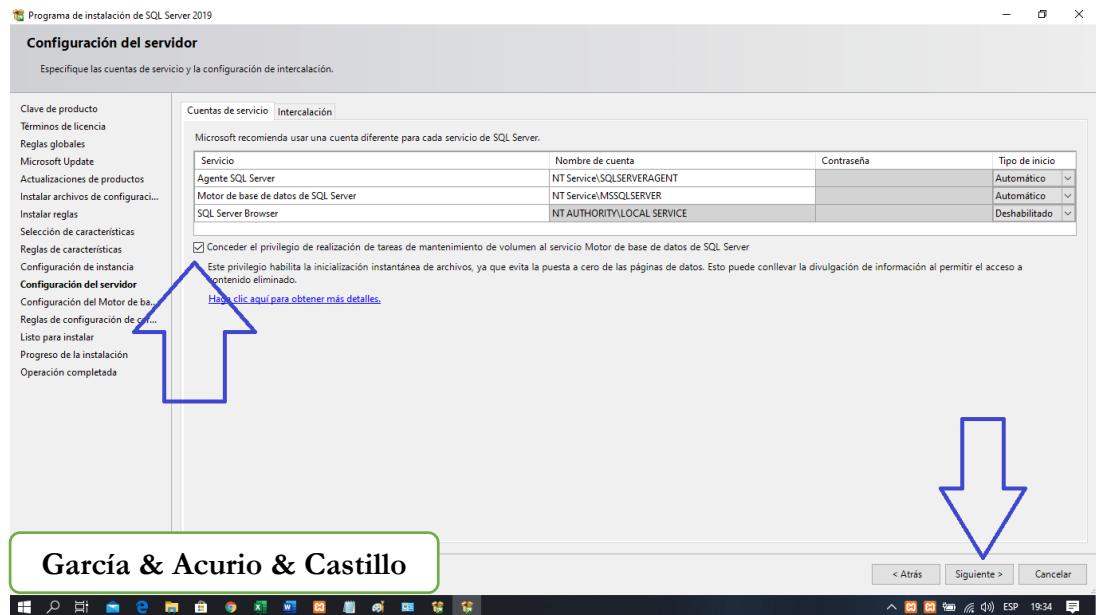
10. Luego aceptamos las características a instalar en el **SQL Server** y le damos aceptar



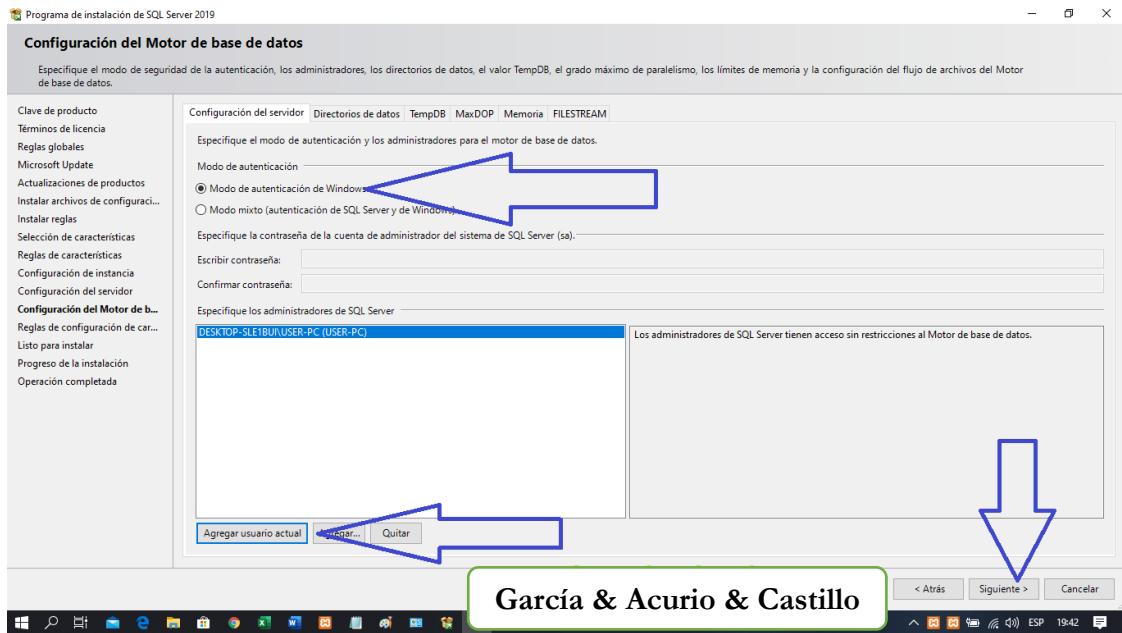
11. luego elegimos un nombre de instancia o le dejamos por default y le damos siguiente.



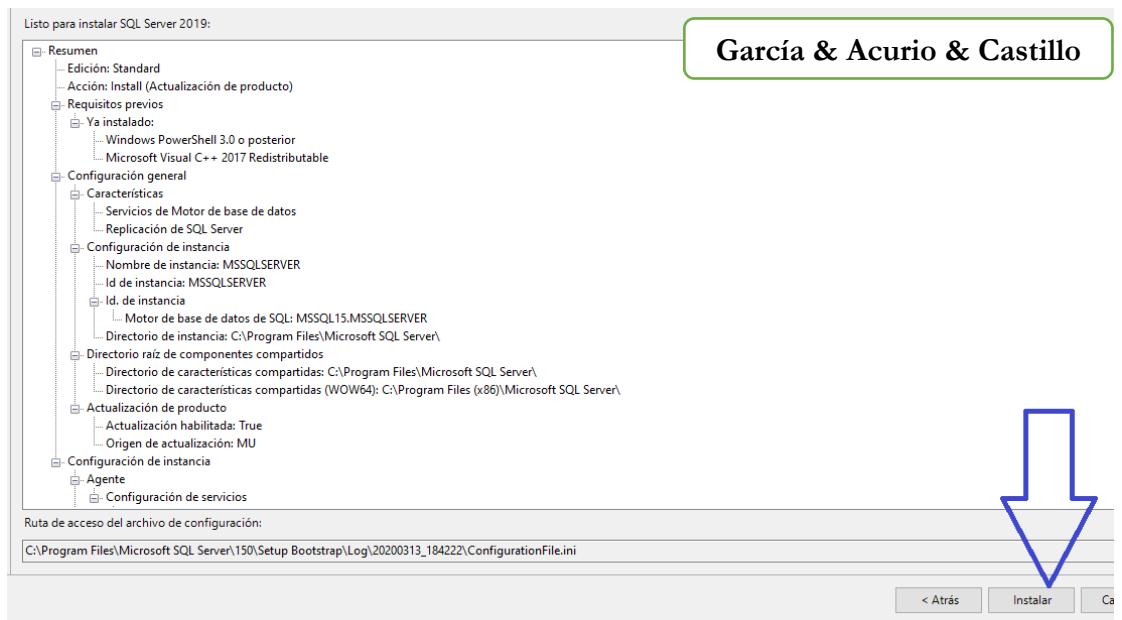
12. luego en la configuración del servidor aceptamos los términos y agente SQL Server automáticamente y damos en siguiente.



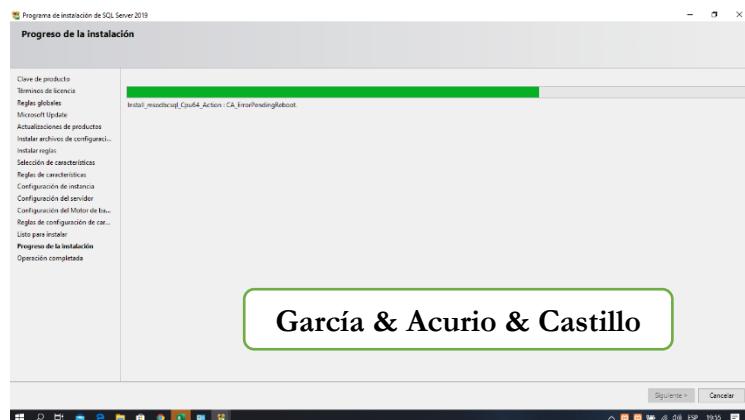
13. luego aquí elegimos el motor de la base de datos y configuramos elegimos un usuario y le damos siguiente.



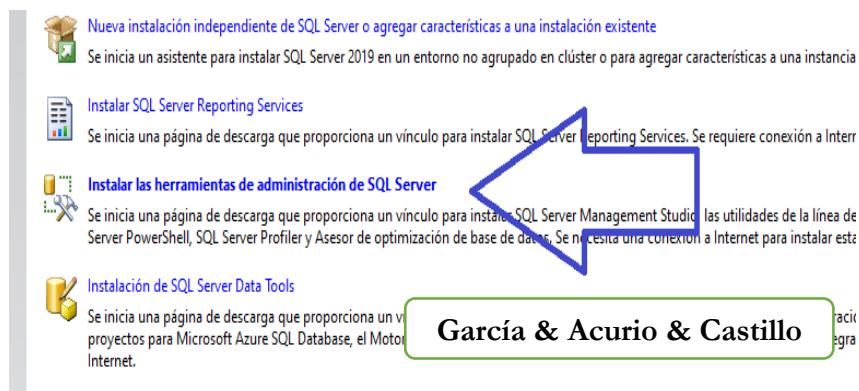
14. luego aquí ya estamos listo para la instalación y vemos que características se van a instalar y damos siguientes.



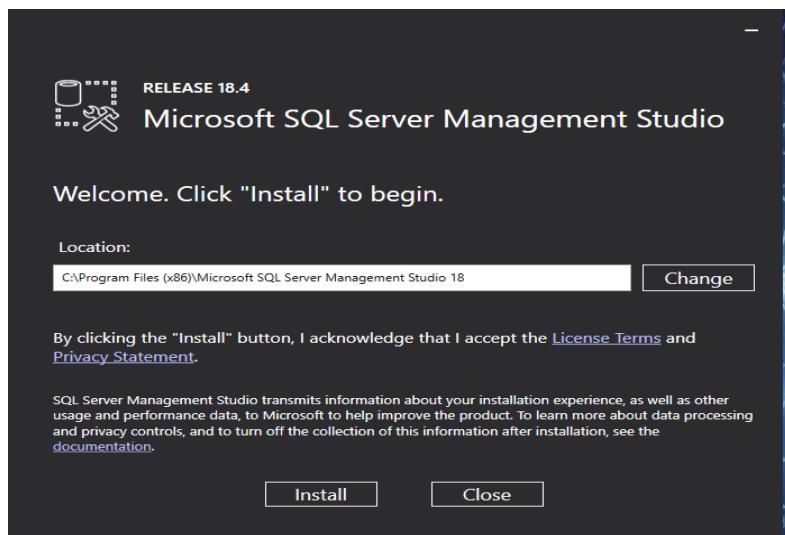
15. luego esperamos q termine la instalación



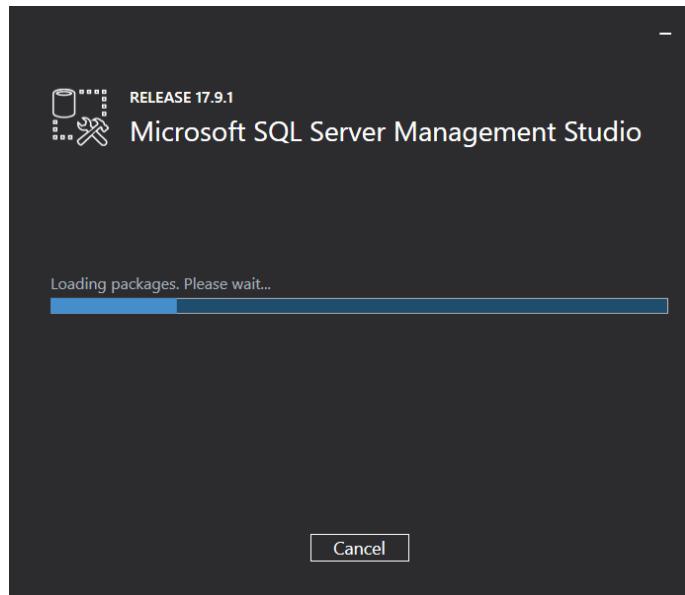
16. Luego una vez instalado eso instalamos las herramientas administrativas de SQL Server.



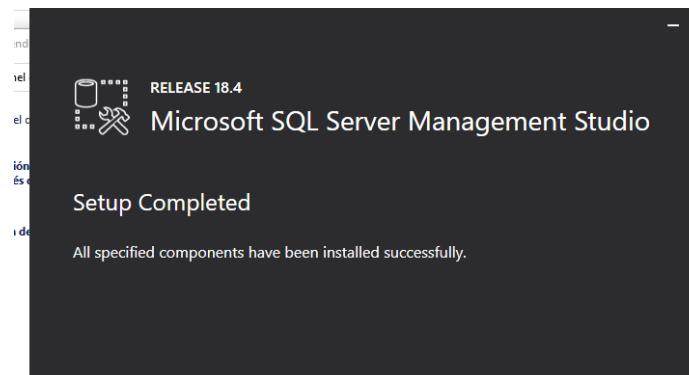
17. Una vez que descargamos la herramienta **Descarga de SQL Server Management Studio (SSMS)** procedemos a instalar e elegimos la ubicación del archivo.



18. Esperamos a que cargue los archivos

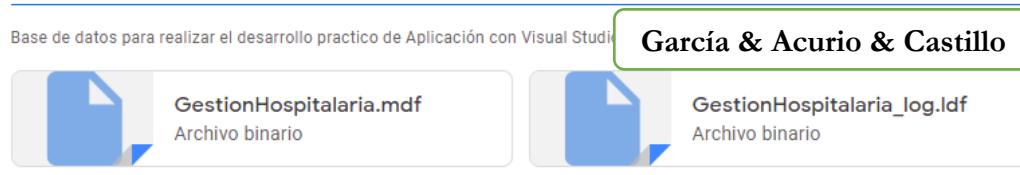


19. Luego de eso ya se instala

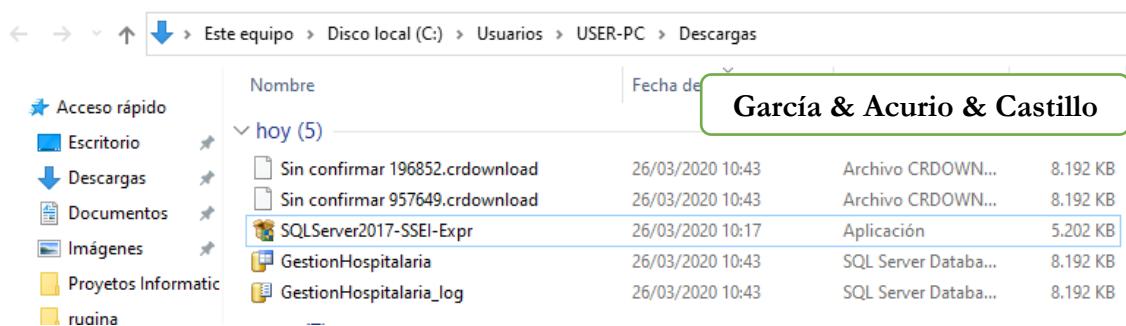


3.2. COPIAR Y PEGAR BASE DE DATOS EN SQL SERVER 2019

20. Primero que debemos hacer es descargar la base de datos Gestión Hospitalaria del classroom



21. Luego lo que tenemos que localizar la base de datos descargada, por lo general se va a descargas



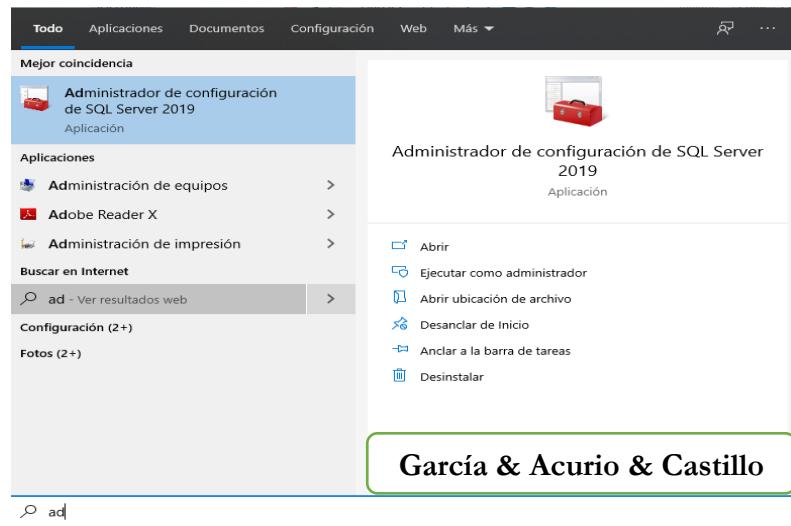
22. Luego nos dirigimos a la dirección donde tienen instalado SQL SERVER si eligieron por defecto la dirección seria “C:\Archivos de Programas \Microsoft SQL Server



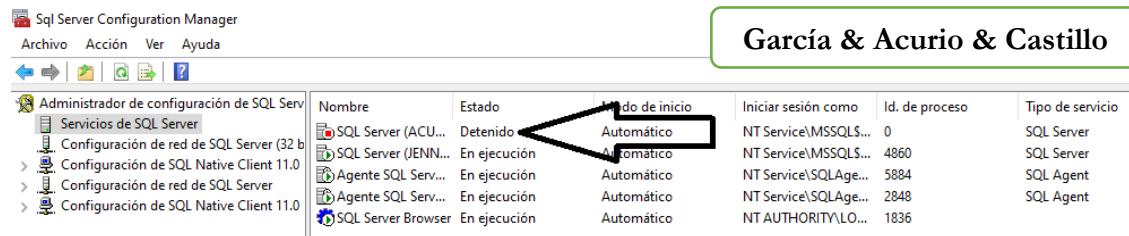
23. Luego estando en esta carpeta buscan la instancia donde quiere copiar la base de datos en este caso se llama “ACURIO” y dirigirse a la dirección “C:\Program Files\Microsoft SQL Server\MSSQL15.ACURIO\MSSQL\DATA



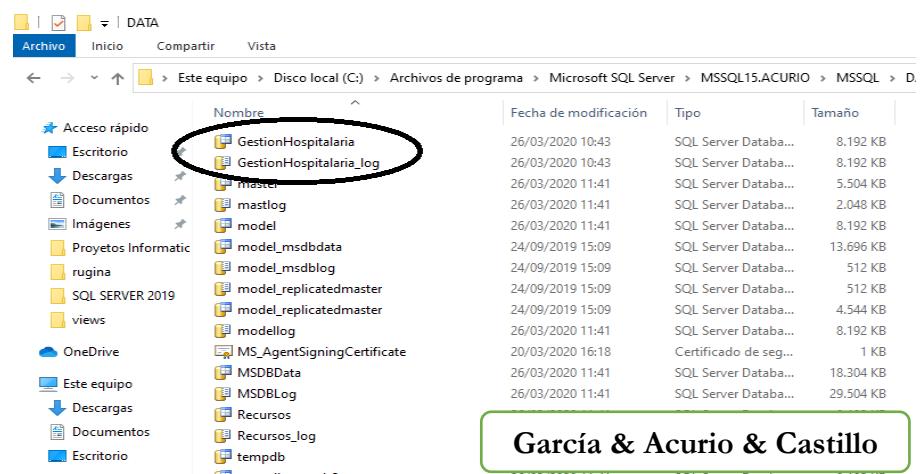
24. Luego se van al paquete que se instalado del motor de SQL y buscan la herramienta “Administrador de configuración de SQL Server 2019”, para detener la ejecución del servidor para poder insertar la base de datos.



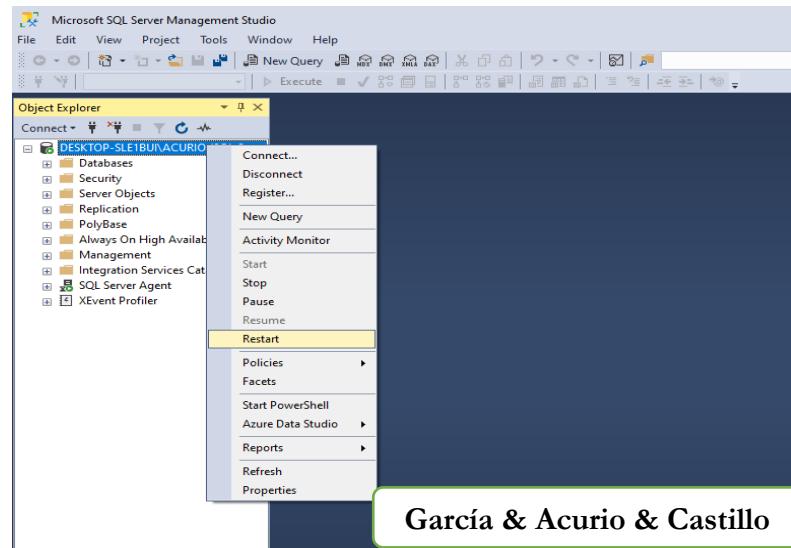
25. Una vez dentro del “Administrador de configuración de SQL Server 2019”, identifican el nombre de la instancia donde quieren copiar la base de datos y cambian su estado a (detenido).



26. Luego de detenido el estado de la instancia, pueden copiar los 2 archivos de base de datos a la instancia seleccionada.

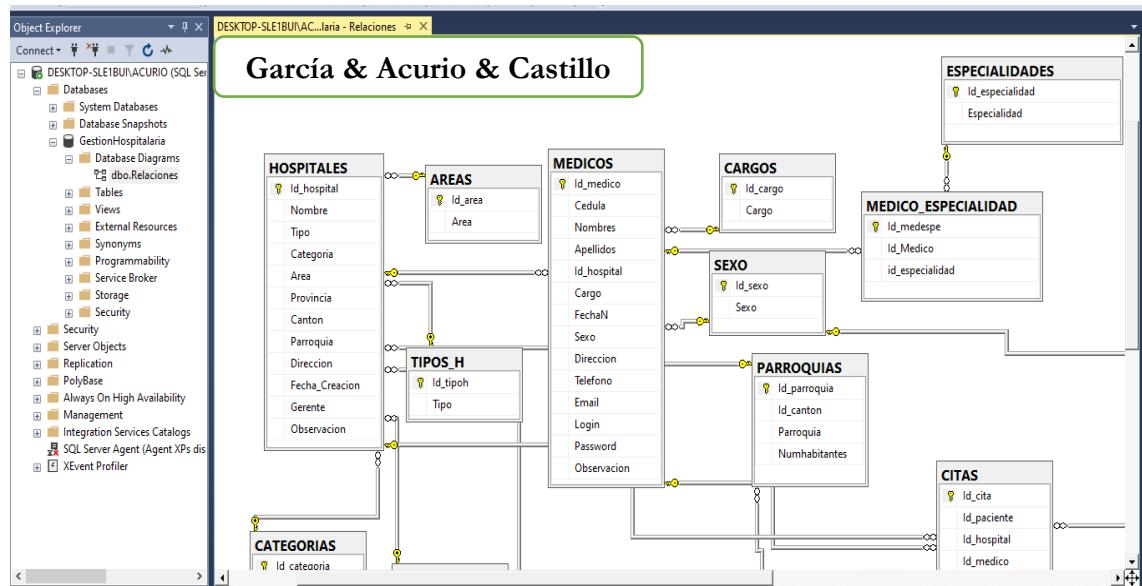


27. Lugo se dirigen al “SQL Server Management Studio” y reinician la instancia hacen clic derecho sobre la instancia y eligen restart.



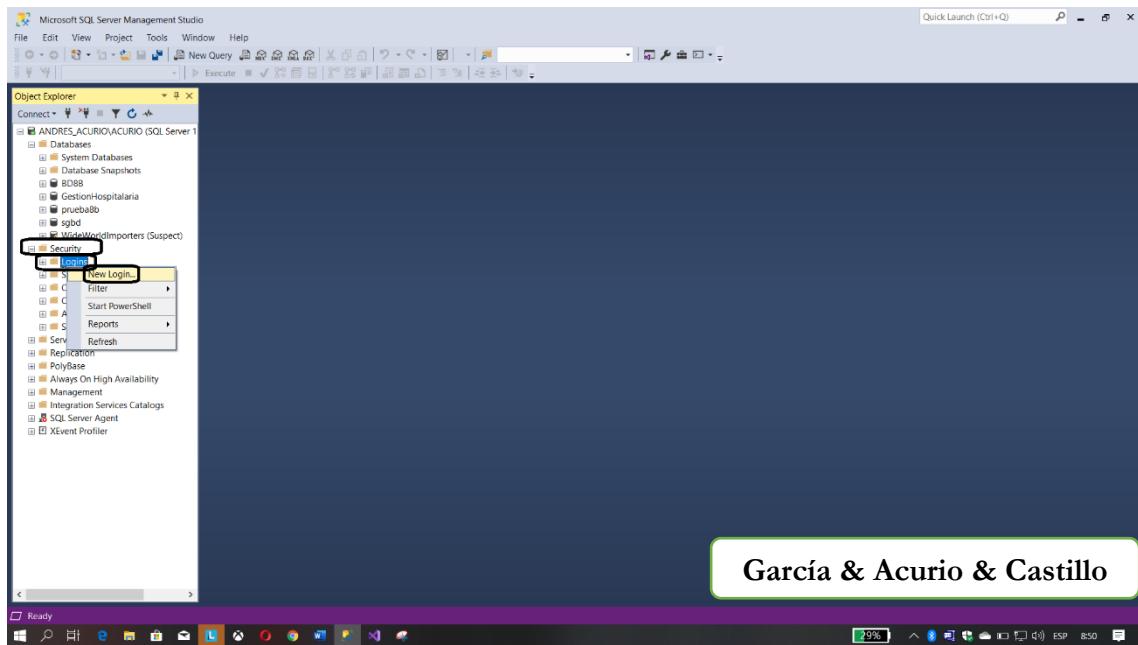
28. Luego una vez ya adentro del Management Studio entramos a la instancia ACURIO vamos a data bases clic derecho attach agregar base de datos Gestión Hospitalaria.

29. Base datos migrada con éxito.

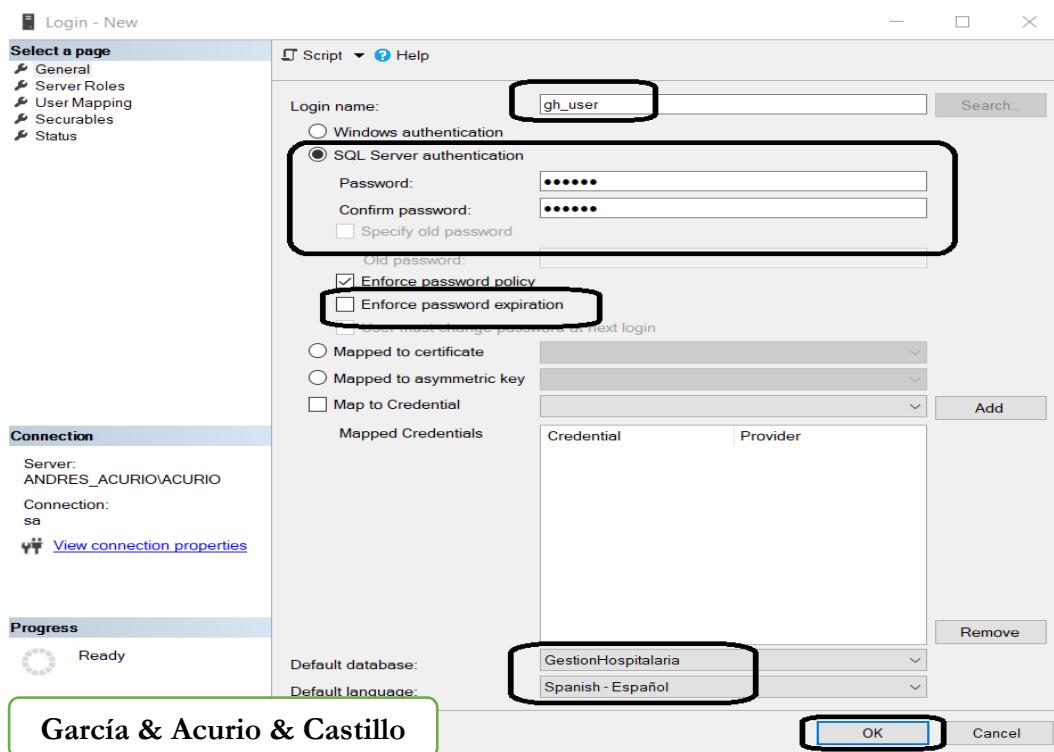


3.3.CREAR Y CONFIGURAR UN LOGIN EN SQL SERVER 2019

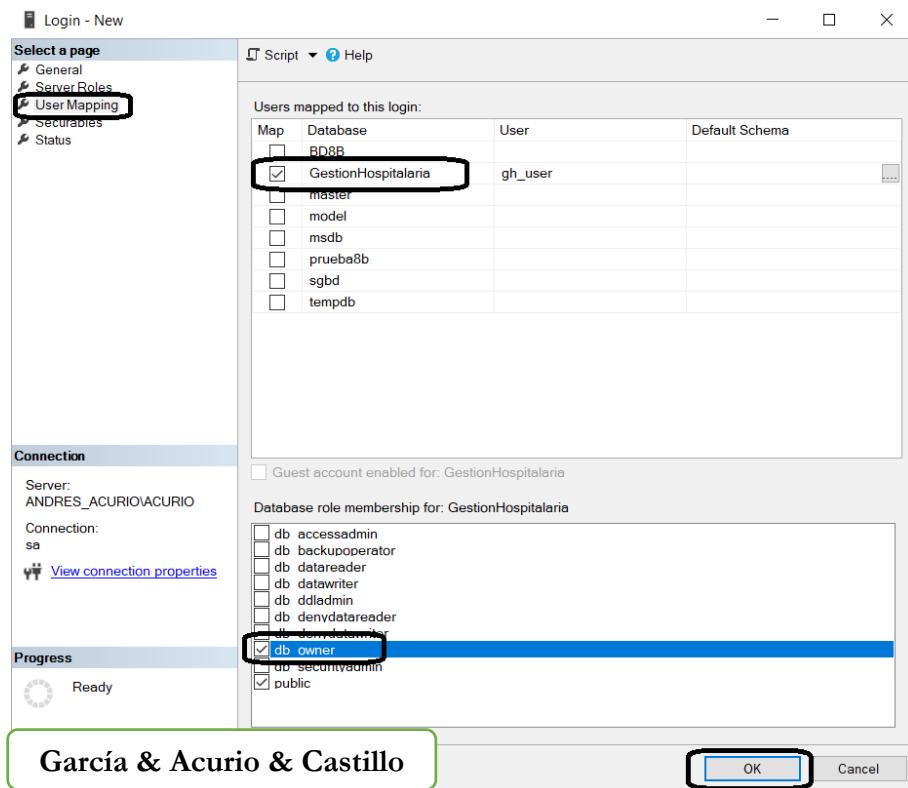
30. Abrimos el SQL SERVER y nos dirigimos a security, luego a Login click derecho new Login



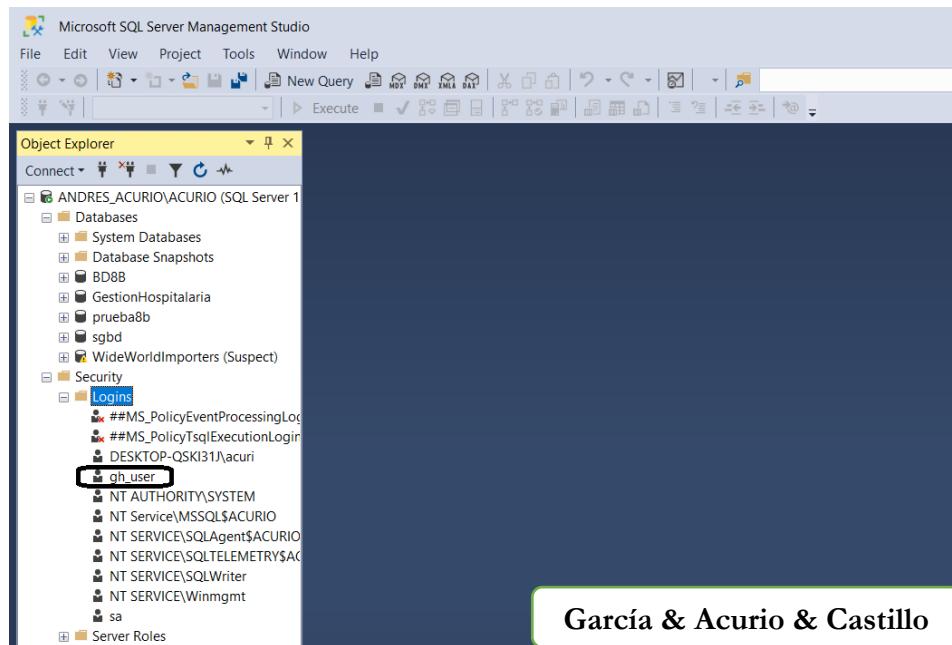
31. Luego se nos despliega una ventana la cual hay que llenar los datos del usuario y contraseña, quitándole la expiración del login, eligiendo la base de datos y el idioma.



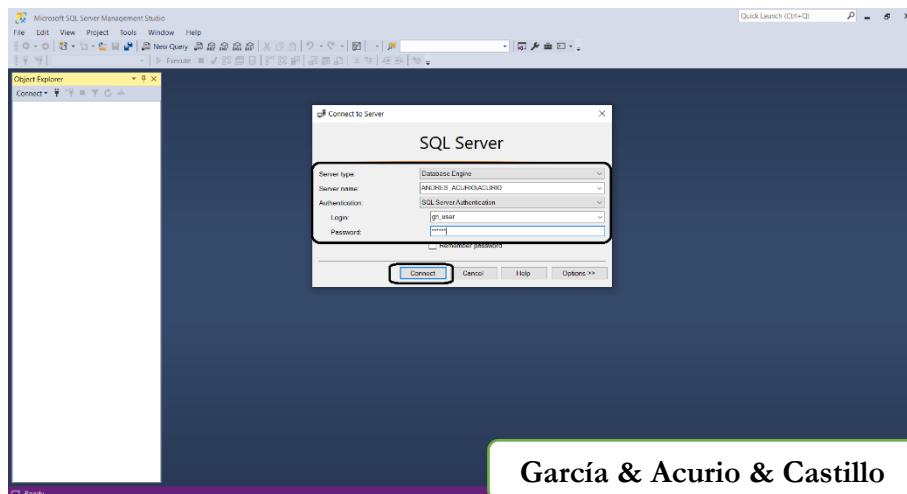
32. Luego le Configuramos la Privacidad.



33. Continuando, se nos Agrega el Usuario

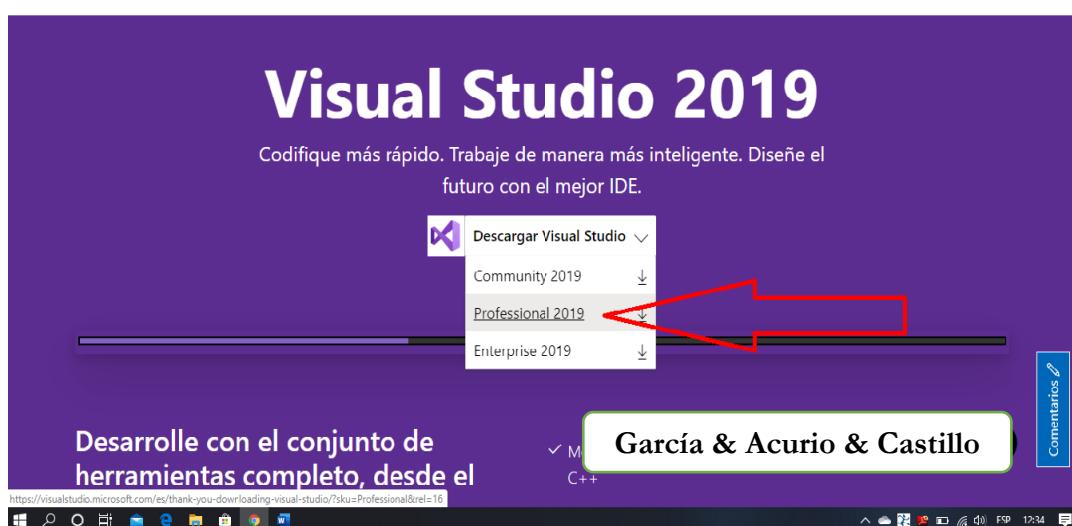


34. Luego Conectamos



3.4. INSTALACIÓN DE VISUAL STUDIO 2019

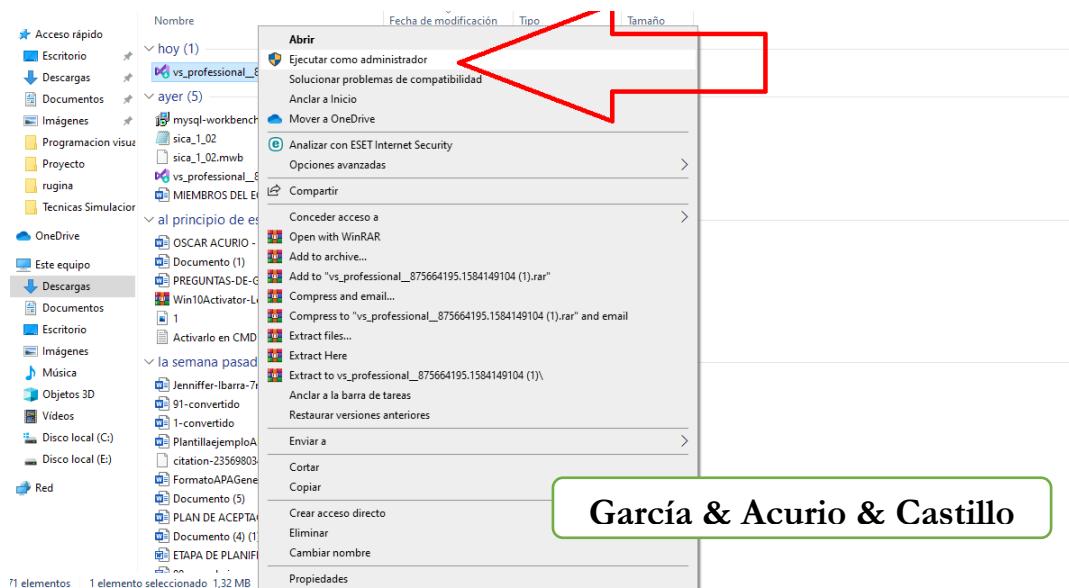
35. Nos dirigimos en Google y digitamos Visual Studio y se nos abre una ventana donde tenemos que elegir qué tipo de visual descargar, elegimos profesional 2019.



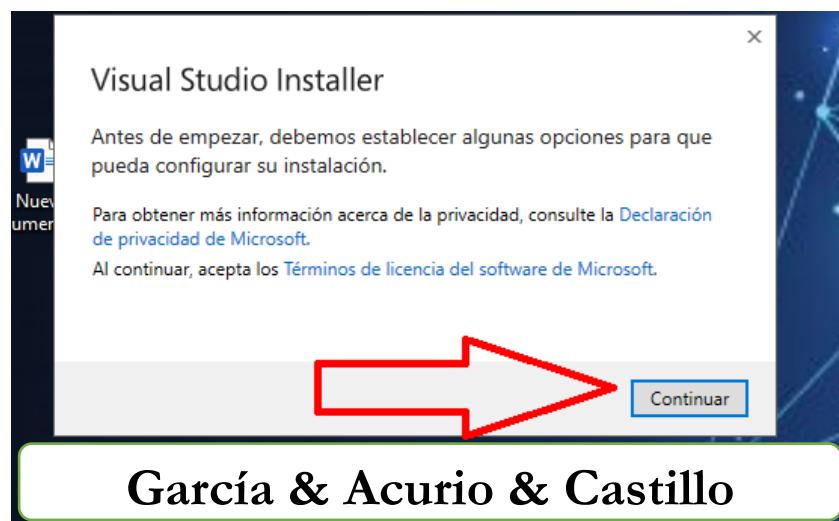
36. Luego se nos descarga un archivo en descargas, nos dirigimos allá



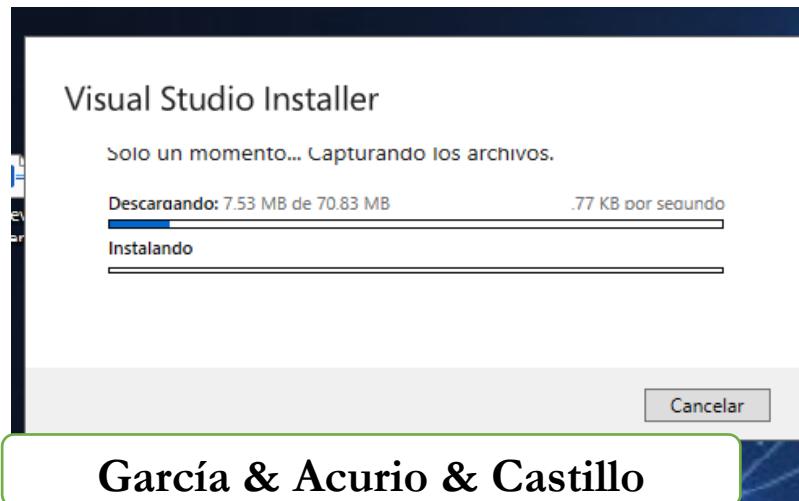
37. luego ejecutamos el archivo como administrador y aceptamos los cambios



38. Luego nos aparece una ventana de Visual Studio Installer le damos clic derecho en la opción continuar. (sabiendo que dando en continuar estamos aceptando los Términos de Licencia de software de Microsoft).

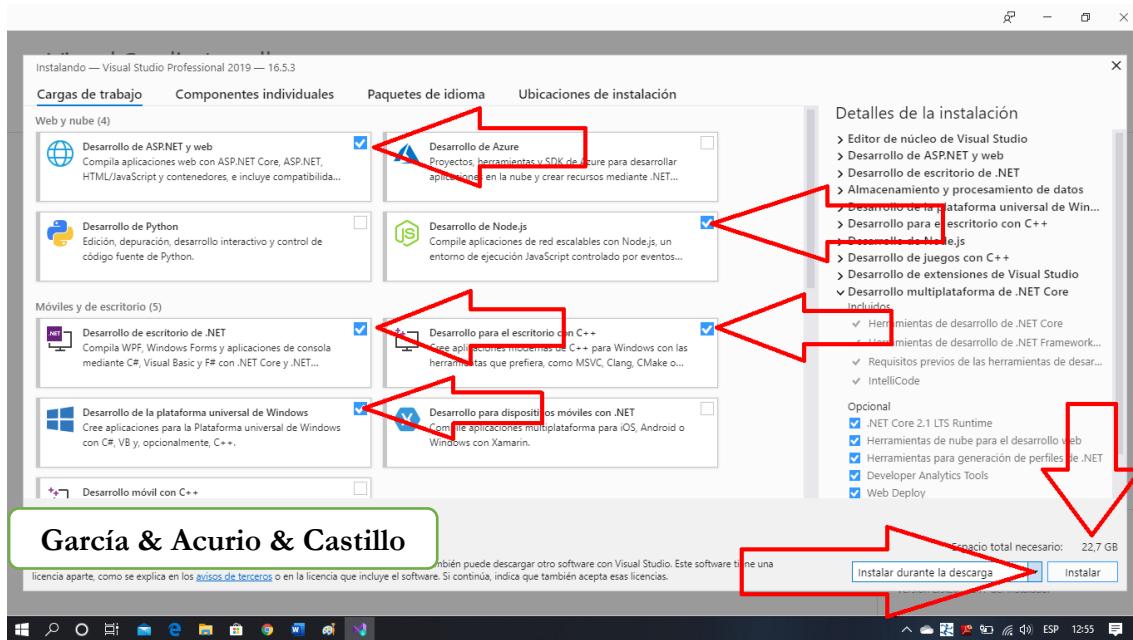


39. Esperamos que se descargue e instale la presentación de Visual Studio

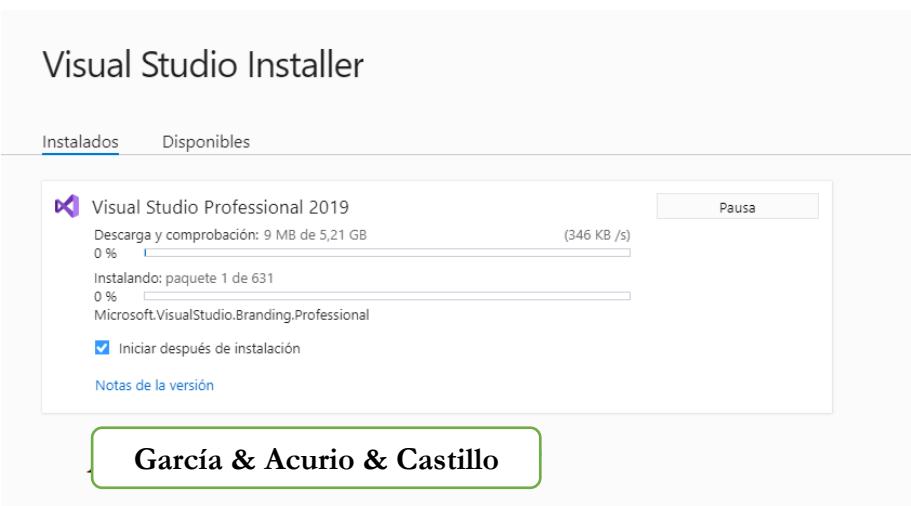


40. Nos emerge una ventana de múltiples opciones elegimos **Cargas de trabajo** aparecen varios componentes elegiremos los más esenciales para lo que utilizaremos en los trabajos de SQL Server y le daremos en instalar.

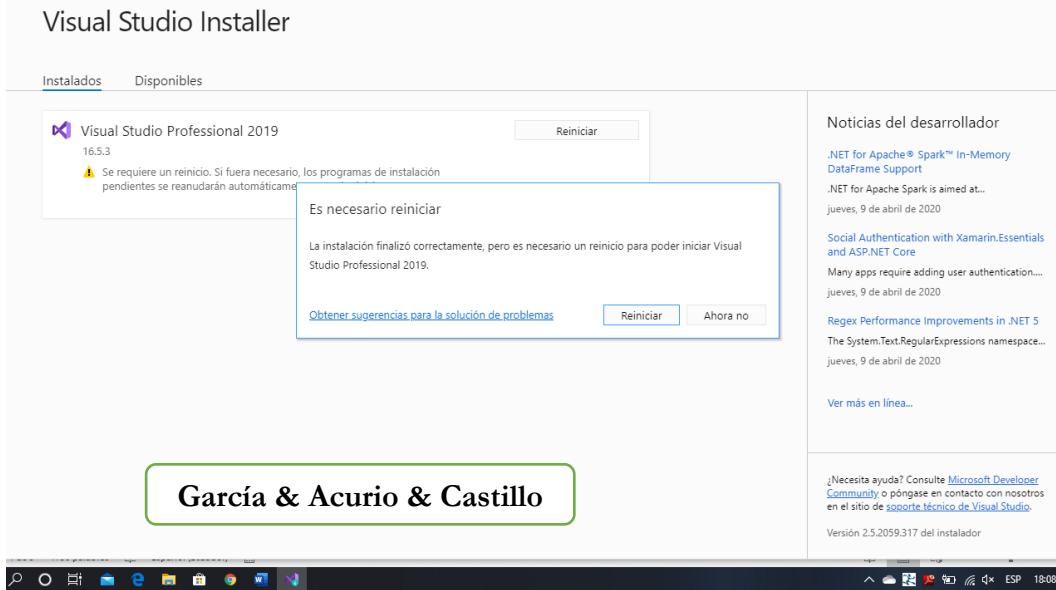
Daremos clic en los 3 componentes que son referente a **Windows**, las herramientas puedan compilar, ejecutar y aportar lo necesario para las aplicaciones que utilizaremos en el sistema operativo, Los que son referentes a **Web y nube**, estas herramientas compilan, desarrollan, conectan lo relativo a las aplicaciones y SQL.



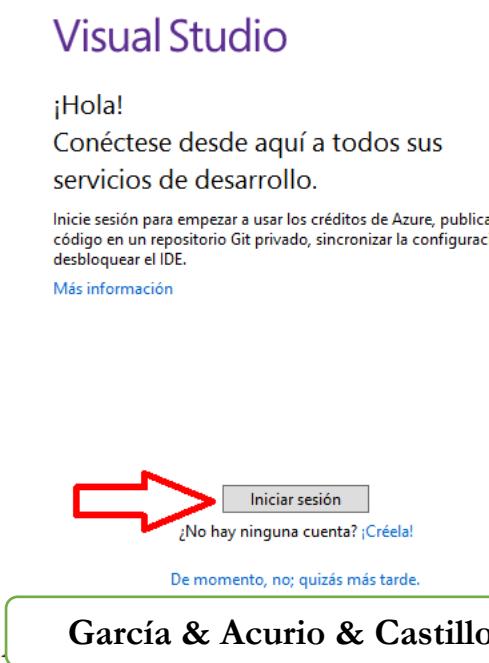
41. Esperamos que realice las operaciones de descarga e instalación de paquetes



42. Terminada la descarga e instalación nos pide que reinicie la máquina para poder visualizar Visual Studio



43. Una vez reiniciado abrimos en inicio Visual Studio 2019 esperamos un momento que cargue, nos da la bienvenida Visual Studio el cual da los tres parámetros si tienes cuenta en Microsoft se da en Iniciar sesión; si no tenemos cuenta elegimos la opción Créela; si la queremos crear la cuenta después elegimos la opción De momento no; quizás más tarde.



44. Después que hemos elegido cualquiera de las 3 opciones nos abre una ventana donde se está preparando Visual Studio.

Visual Studio

La preparación está en curso

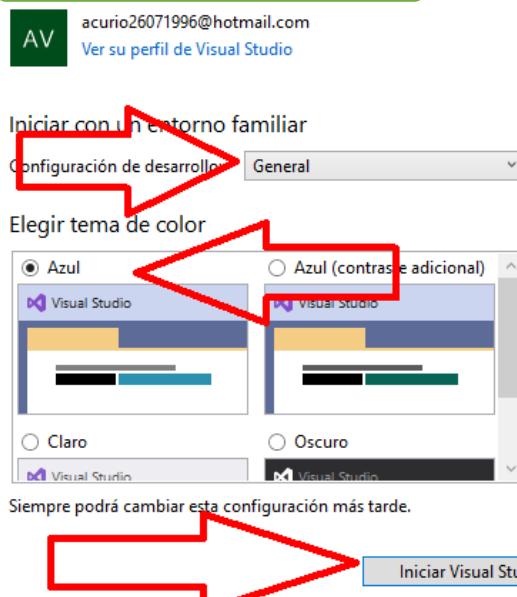
Gracias por su paciencia.

García & Acurio & Castillo

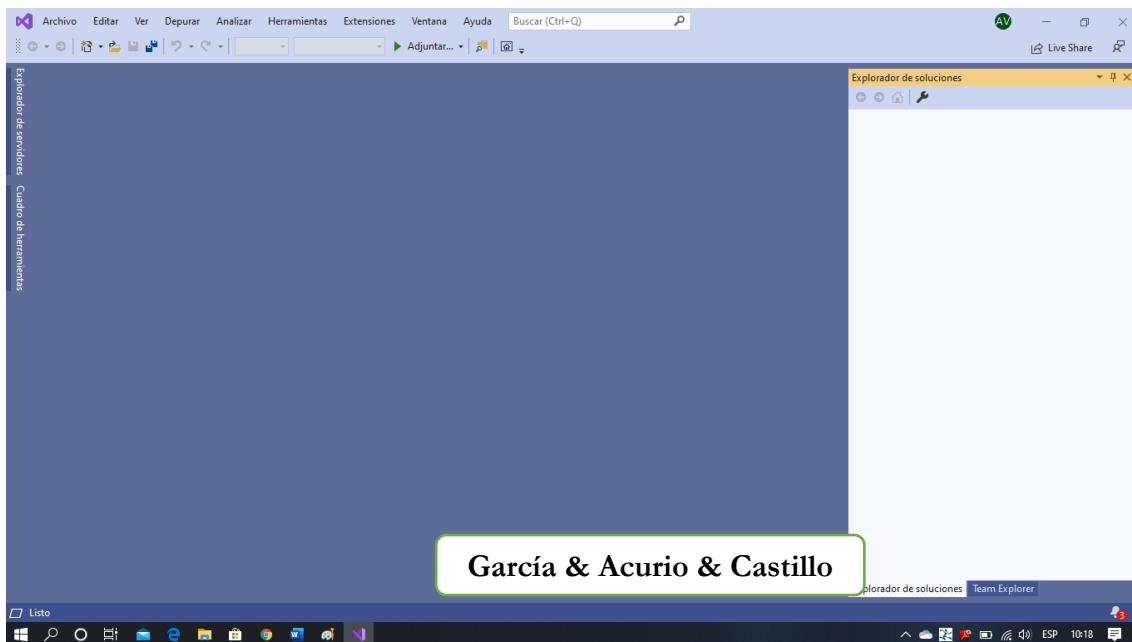
45. Luego nos abre una ventana de como deseamos iniciar Visual Studio definimos configuración de desarrollo y tema de color, pasamos luego a Iniciar Visual Studio.

Visual Studio

García & Acurio & Castillo



46. Nos presenta la venta la página de **inicio – Microsoft Visual Studio** donde trabajaremos en la interfaz gráfica, realizaremos múltiples tareas.



3.5. CREACIÓN DE UN PROYECTO EN VISUAL STUDIO 2019

47. Abrir **VISUAL STUDIO 2019** y Crear un Nuevo Proyecto

Visual Studio 2019

Abrir recientes

Cuando abra proyectos, carpetas o archivos en Visual Studio, se mostrarán aquí para obtener un acceso rápido.

Puede arrancar cualquier elemento que abra con frecuencia para que aparezca siempre en la parte superior de la lista.

Tareas iniciales

Clonar un repositorio

Obtiene código desde un repositorio en línea, como GitHub o Azure DevOps.

Abrir un proyecto o una solución

Abre un archivo .sln o proyecto de Visual Studio local.

Abrir una carpeta local

Navegar y editar el código en cualquier carpeta

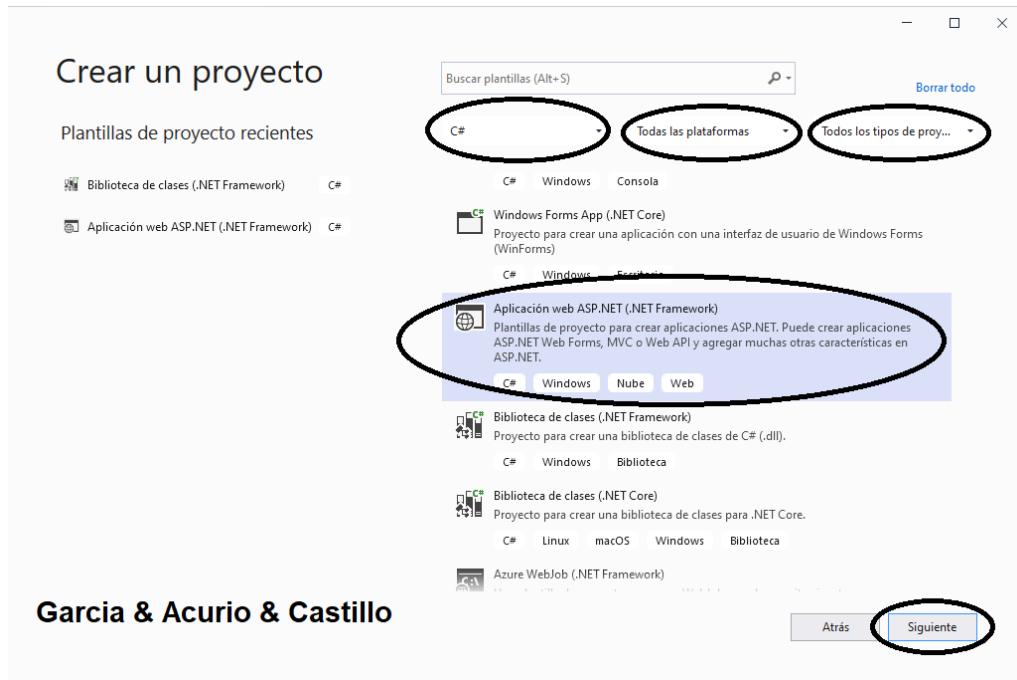
Crear un proyecto

Elija una plantilla de proyecto con la técnica scaffolding de código para comenzar.

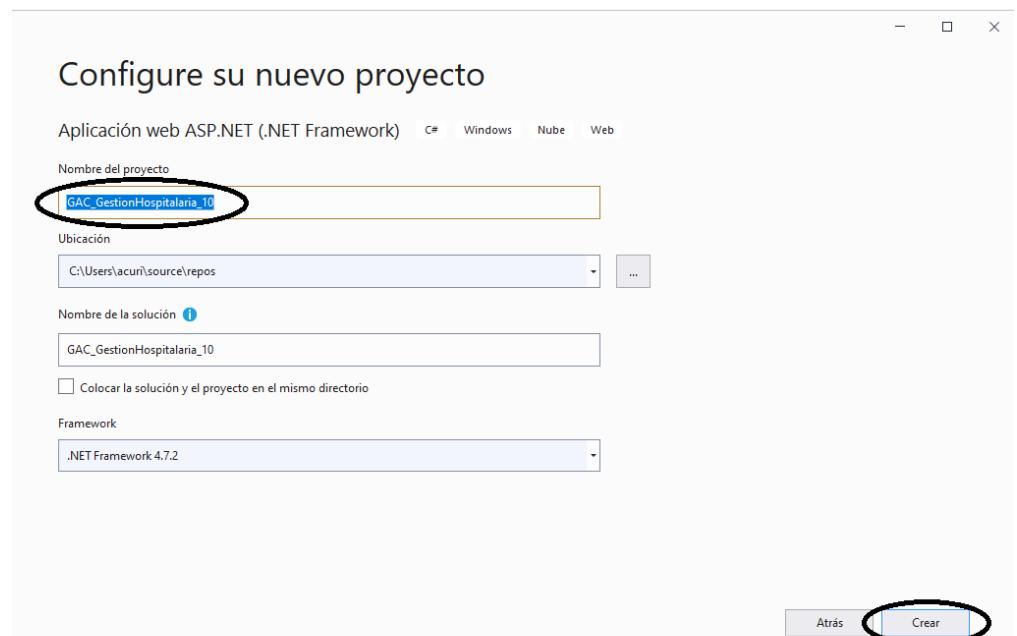
[Continuar sin código →](#)

Garcia & Acurio & Castillo

48. Una vez en la ventana crear un proyecto, elegimos la opción “C#” y tipo de proyecto “Aplicación web ASP.NET. (Net Framework)” y le damos en siguiente.



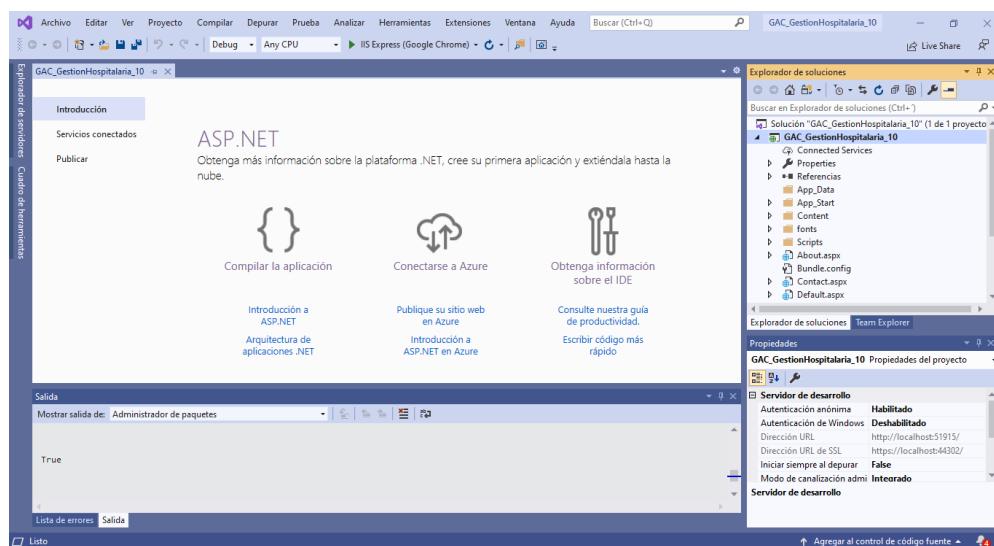
49. Luego elegimos el nombre del Proyecto y le damos en Crear



50. Luego en la página crear una aplicación **Web ASP.NET**, aquí elegimos la opción “**Web Forms**” y Crear

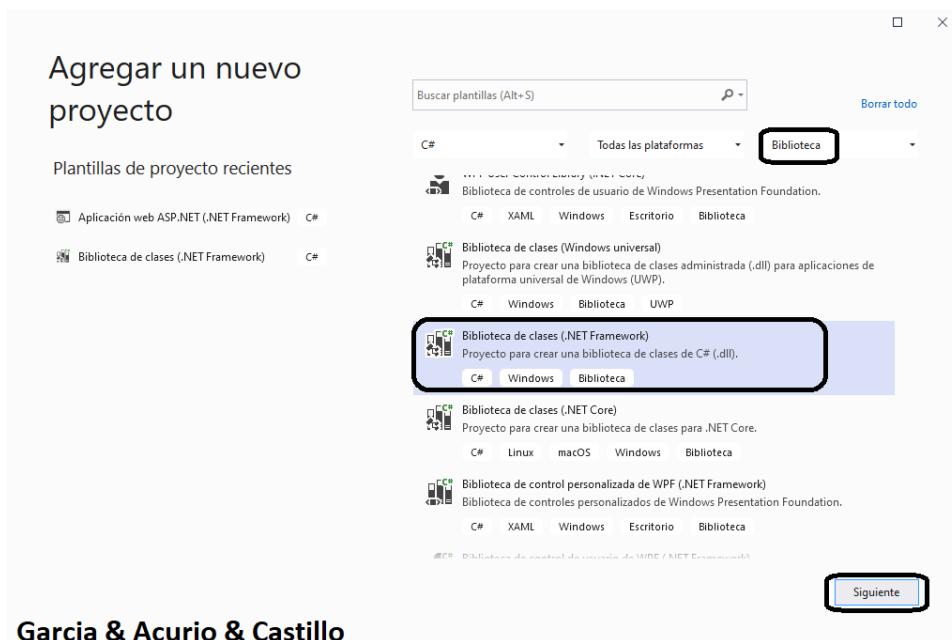


51. Luego de Crear el programa se crea un sinnúmero de cosas, este proceso demora un poco, y por último debe salir una ventana idéntica a la que se muestra en la imagen.

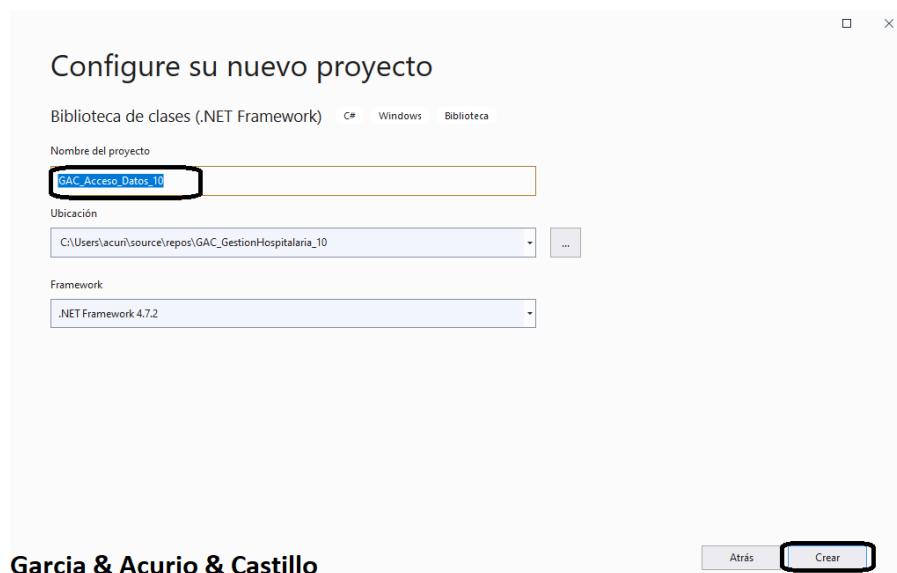


3.6. CREACIÓN DE LAS CAPAS

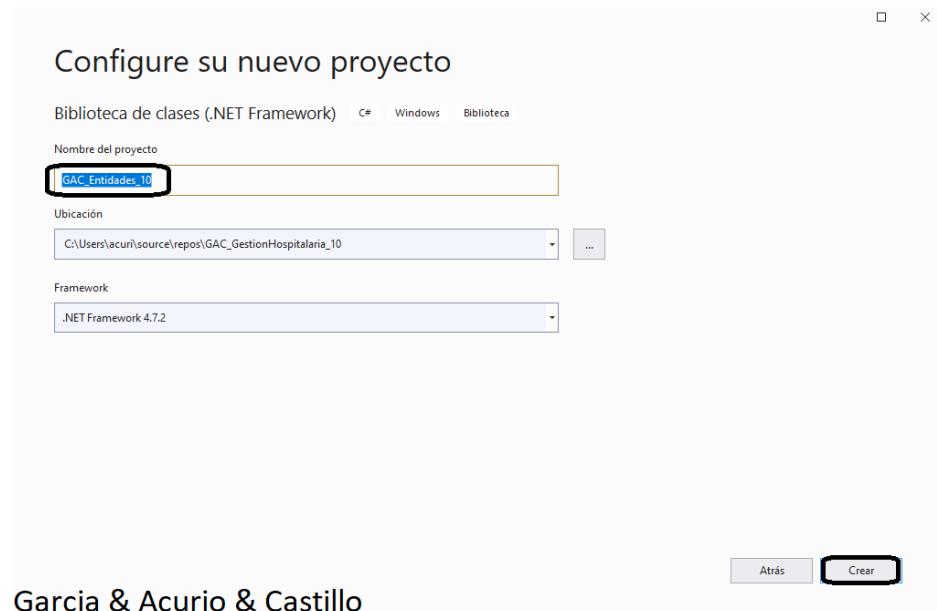
52. Para la creación de capas hacer Clic derecho sobre la solución, **agregar-nuevo** proyecto, Y elegimos “**Biblioteca de clases. Net Framework**” y le damos en siguiente



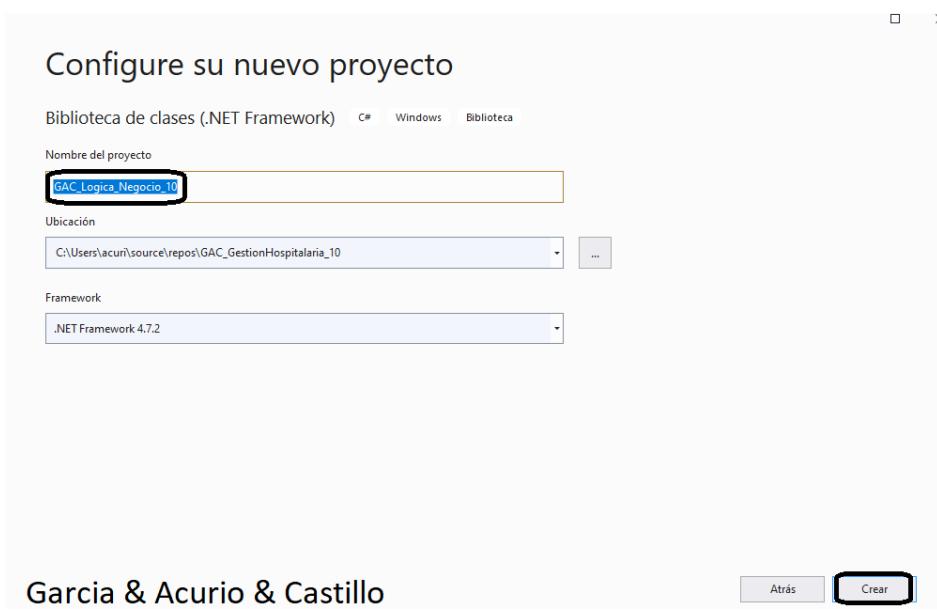
53. Una vez dado siguiente a ese nuevo proyecto le colocare el nombre “**GAC_Acceso_Datos_10**”, que representara a la **capa de datos**, y le damos en crear



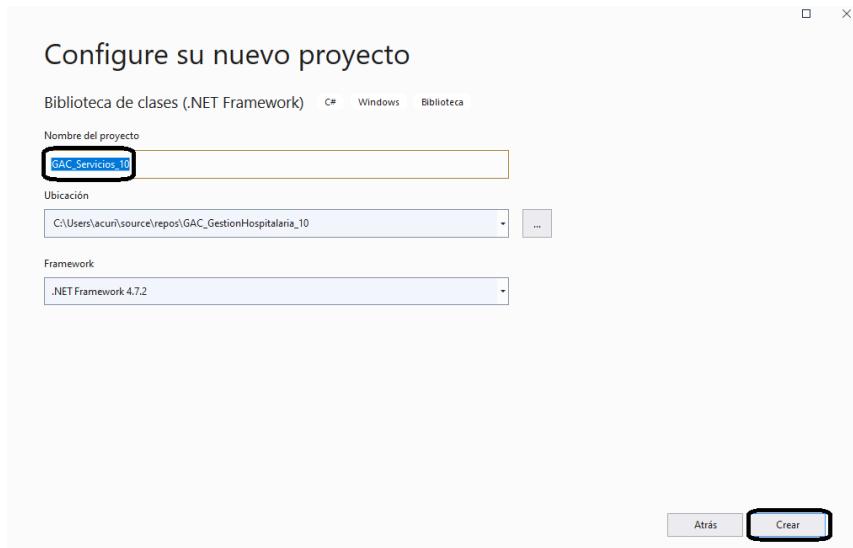
54. Hacemos el mismo paso para crear las otras capas. Creación Capa de Entidad donde le colocare el nombre “**GAC_Entidades_10**” y le damos en crear



55. Hacemos el mismo paso para crear las otras capas. Creación Capa Lógica donde le colocare el nombre “**GAC_Logica_Negocio_10**” y le damos en crear

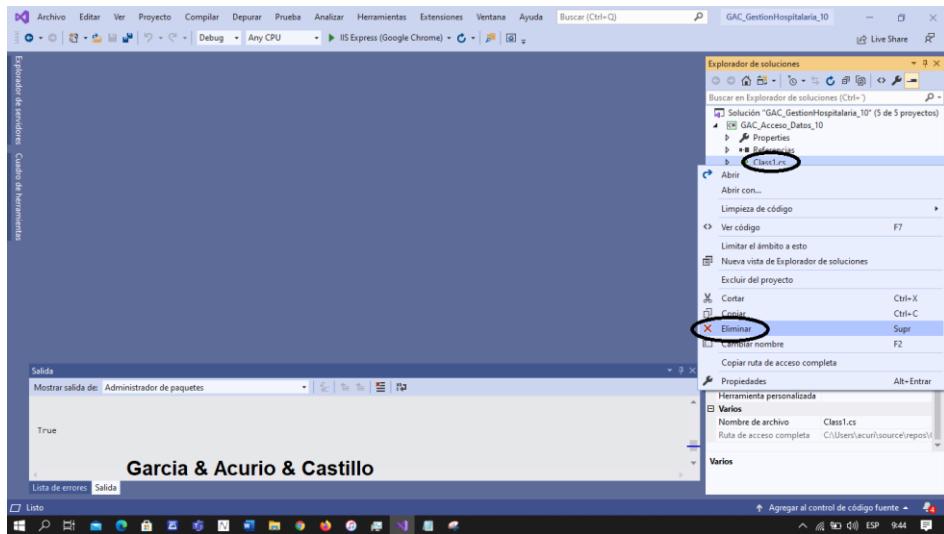


56. Hacemos el mismo paso para crear las otras capas. Creación Capa de Entidad donde le colocare el nombre “**GAC_Servicios_10**” y le damos en crear



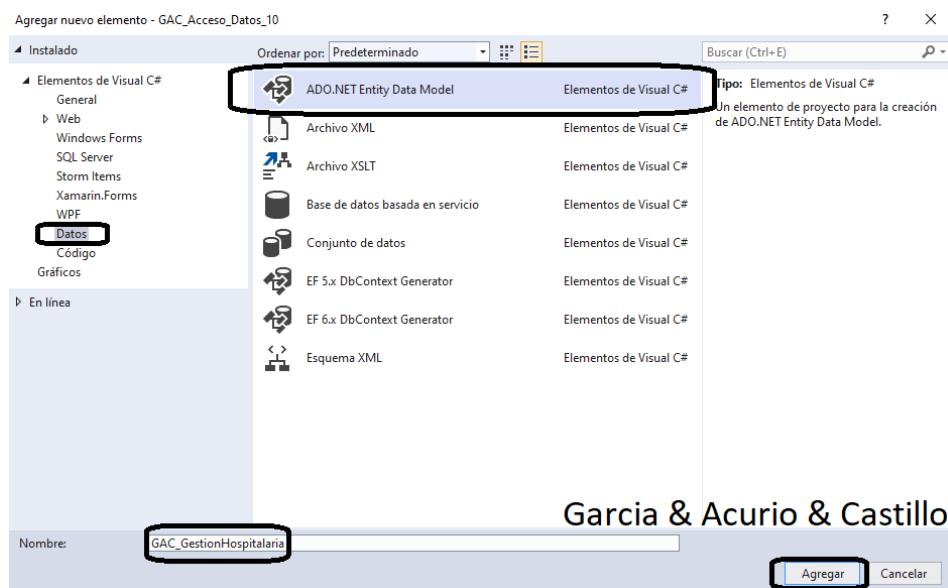
Garcia & Acurio & Castillo

57. Una vez creado todas las capas procedemos a eliminar un archivo class.cs que se crean dentro de cada que creamos anteriormente

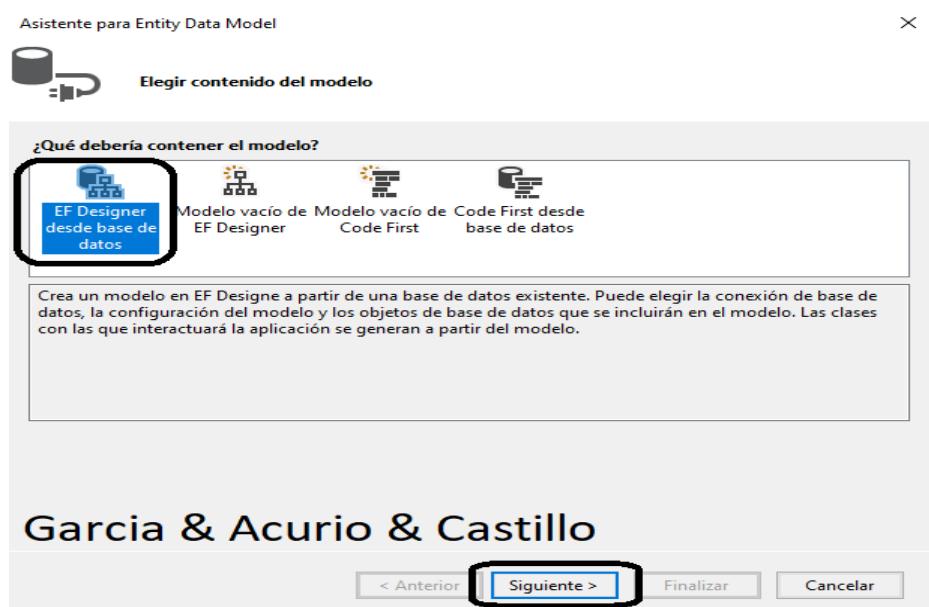


3.7. CONEXIÓN DE LA CAPA DE DATOS CON LA BASE DE DATOS

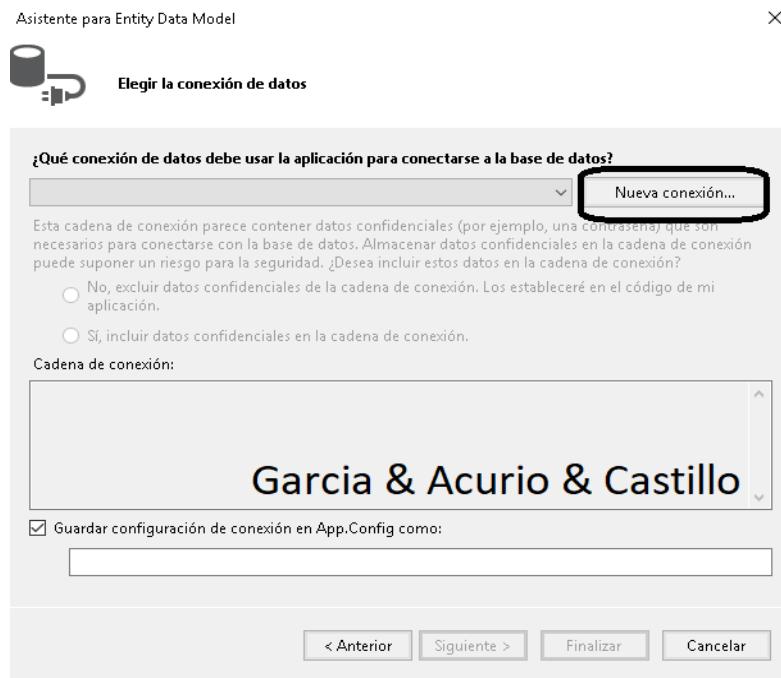
58. Dirigirse a la capa de datos y hacer clic derecho sobre esta y agregar “nuevo elemento”-opción “Data”- “ADO.NET Entity Data Model” y le colocan el nombre que le pusieron a la solución en este caso “GAC_GestionHospitalaria” y le damos en agregar



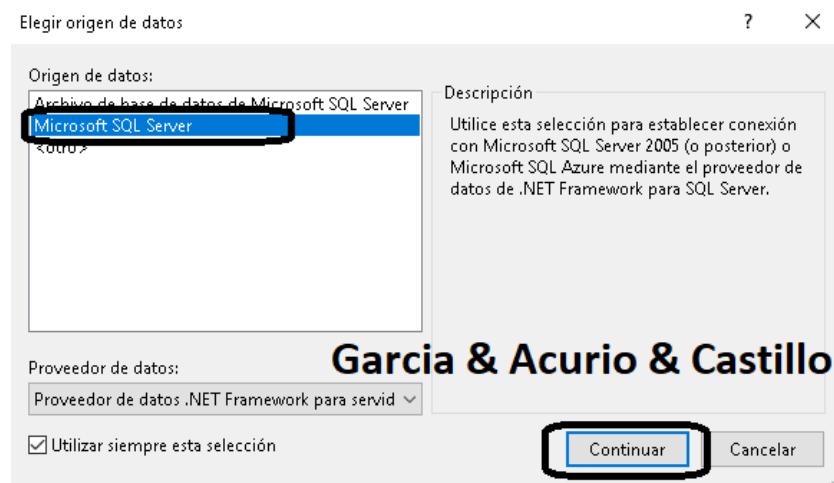
59. Aparecerá una nueva ventana escogen la primera opción “EF Designe” y luego dar en siguiente.



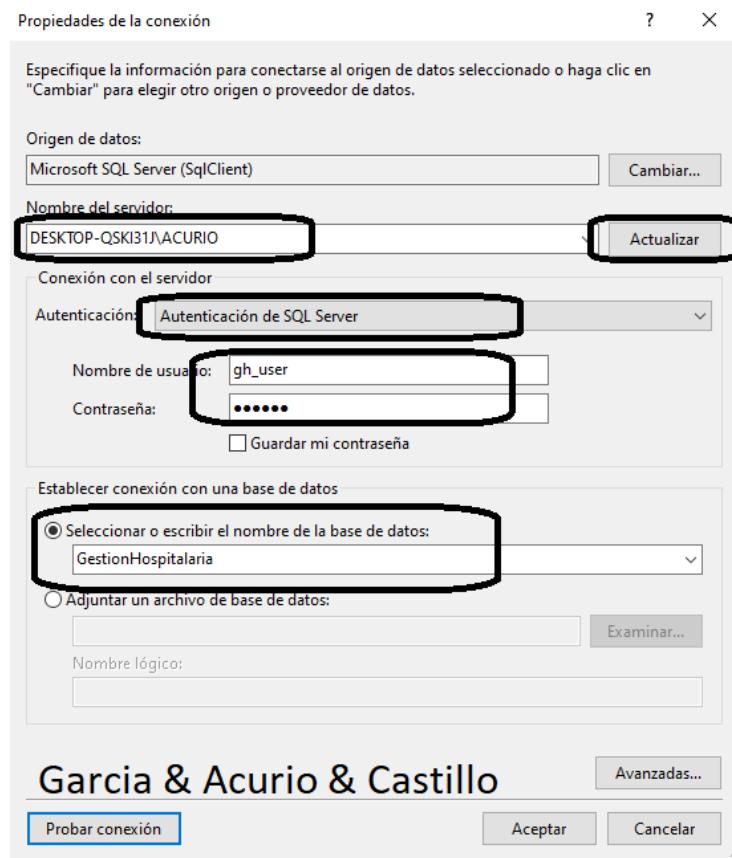
60. Luego de dar en siguiente aparecerá la Ventana de Asistente para Entity Data Model, eligen la opción “nueva conexión”.



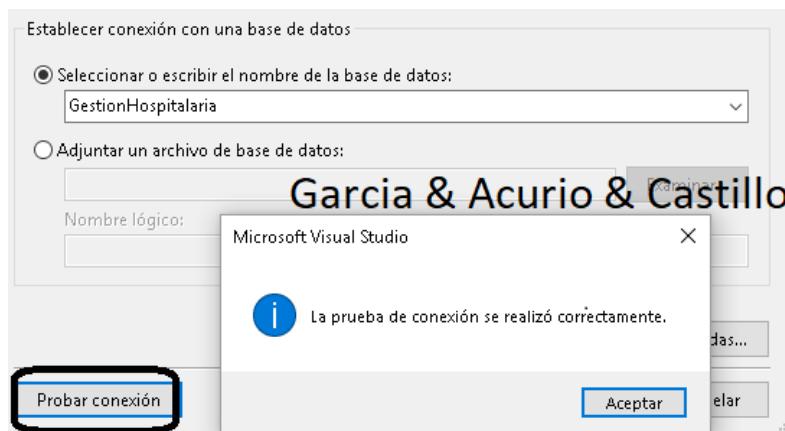
61. Luego aparece una nueva ventana, en esta eligen la primera opción “Microsoft SQL Server”, y le dan en continuar



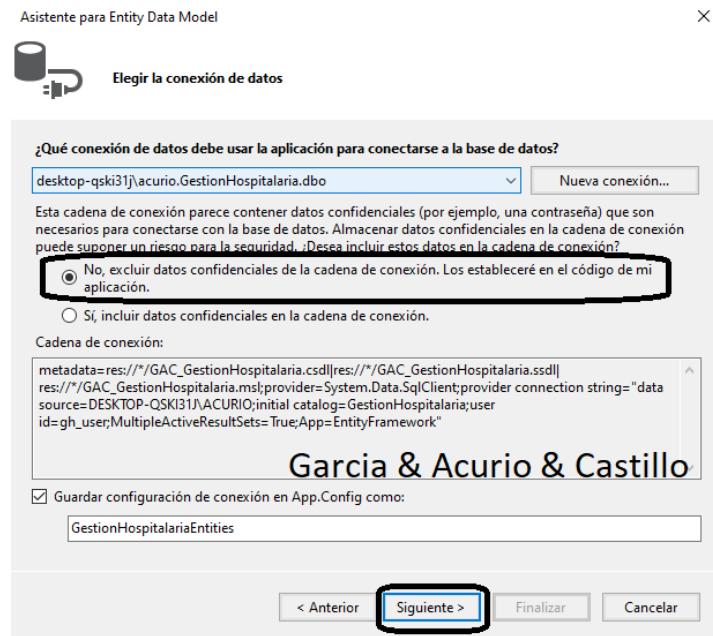
62. En la nueva ventana, le dan en **Actualizar** y esperan a que termine y despliegan la flecha hacia abajo para visualizar los **servidores** y eligen uno, colocan sus credenciales y además seleccionan la base de datos.



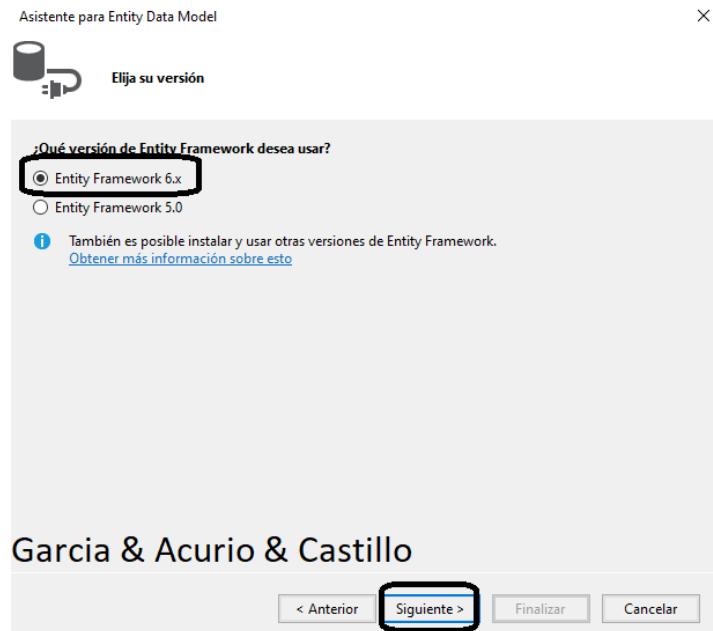
63. Luego probamos la conexión y le damos en aceptar



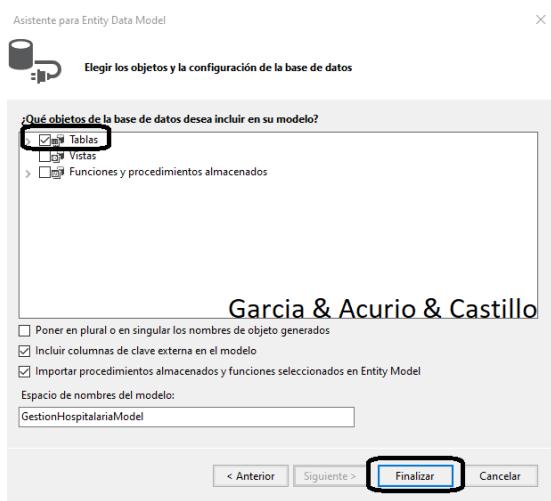
64. Luego le aparecerá otra ventana, en esta marcan la opción “**no**” Y luego dan en siguiente



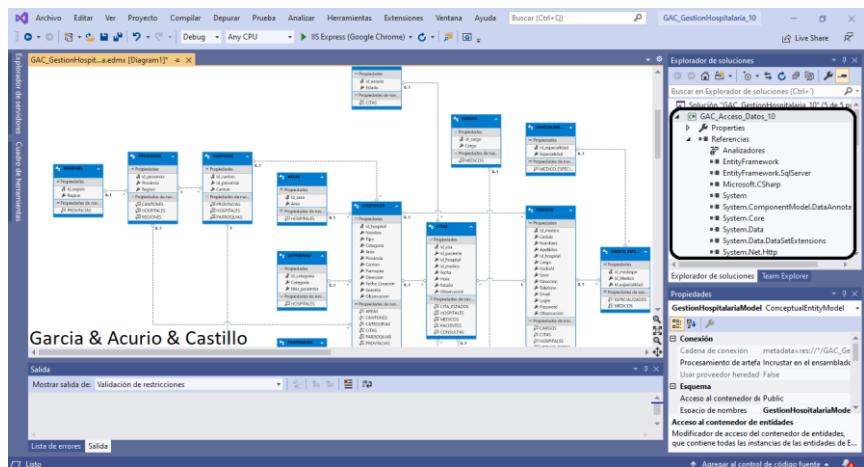
65. Luego en la ventana de elección de la versión de “Entity Framework” eligen la última versión “6.x”, y luego dar en siguiente.



66. Luego en la ventana de elección de objeto y configuración de base datos, en esta ventana eligen la opción “**Tablas**” que es solo lo que vamos a necesitar y luego dar en finalizar

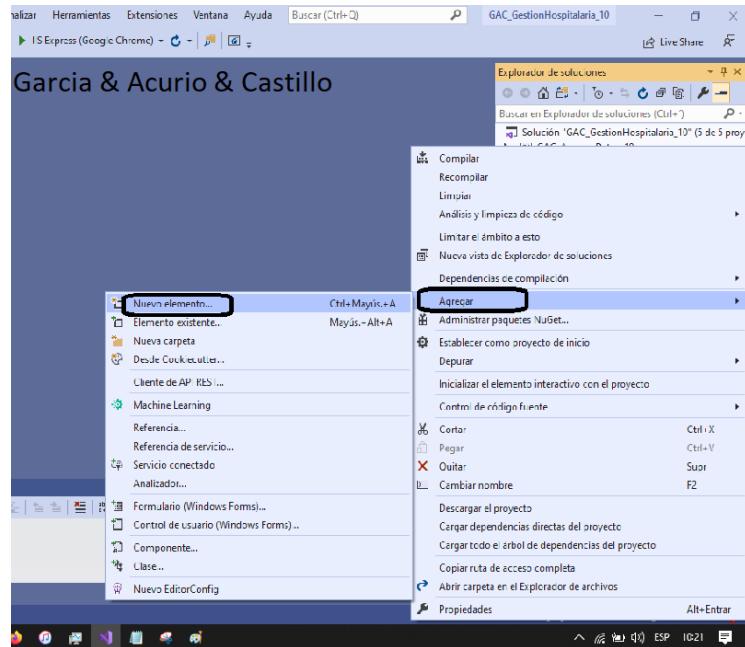


67. El programa hará un sinnúmero de procesos, después de eso nos entregará el diagrama completo de la base de datos sin haber digitado código nada.

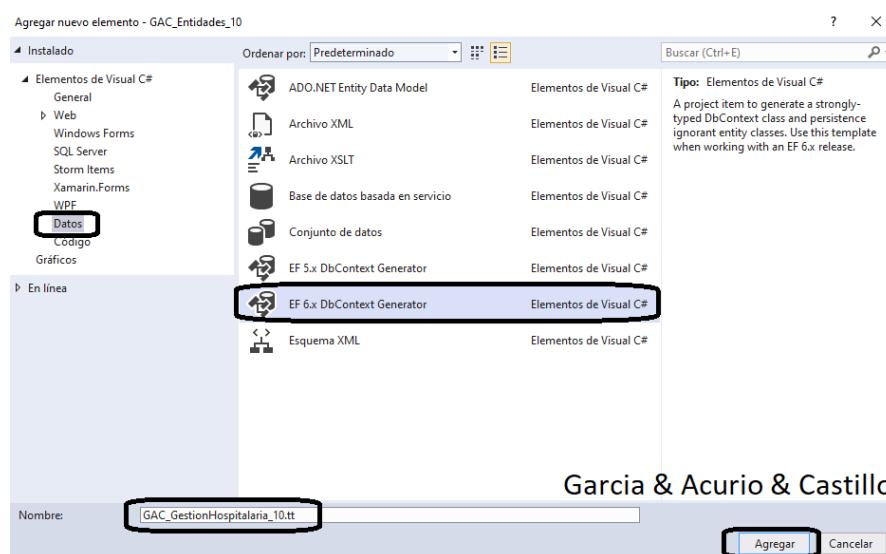


3.8. REFERENCIAR LA CAPA DE DATOS CON LA CAPA ENTIDADES

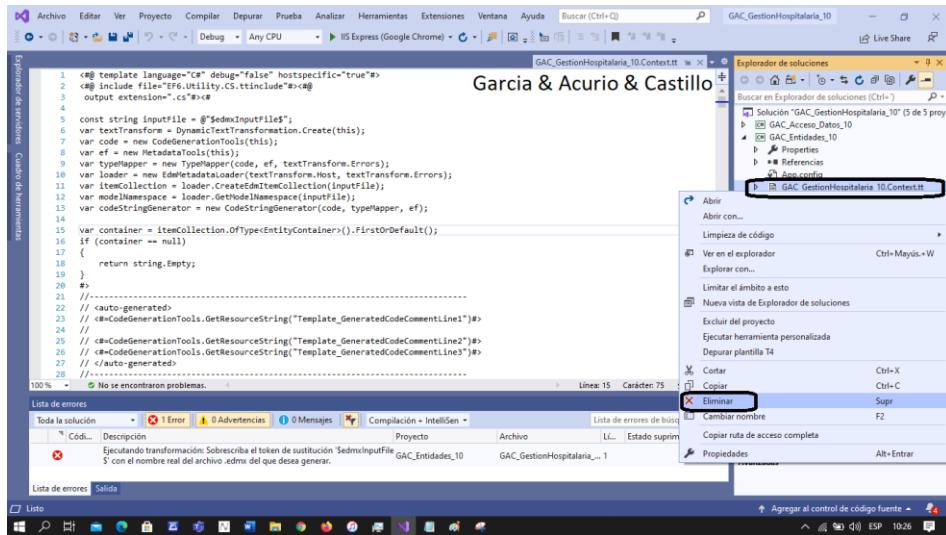
68. Hacer clic derecho sobre la Capa Entidades y elegir la opción **agregar-nuevo**.



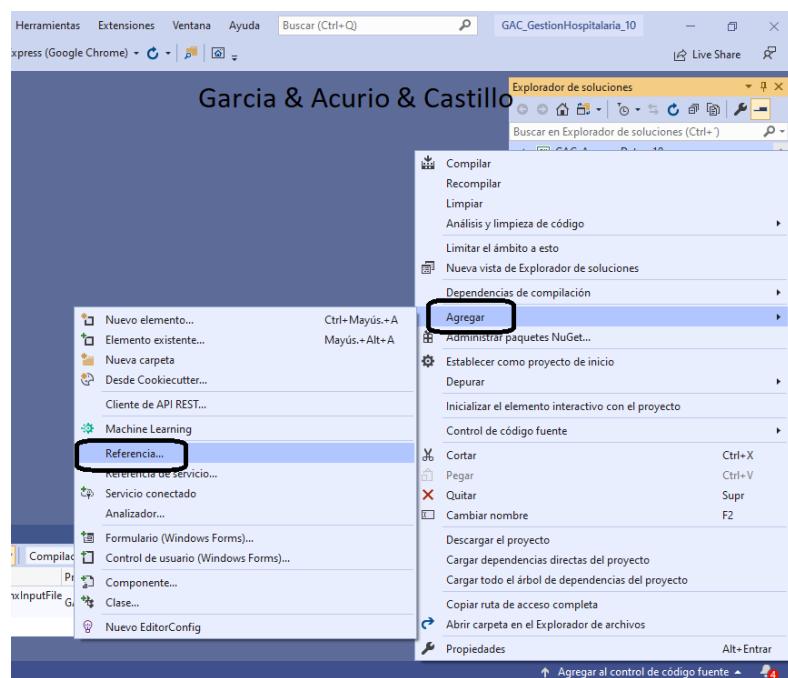
69. Luego en la venta elegimos **Datos** y “**EF 6.x DbContext Generator**”, y colocar el mismo nombre que tiene la solución en este “**GAC_GestionHospitalaria_10**” y le damos en agregar.



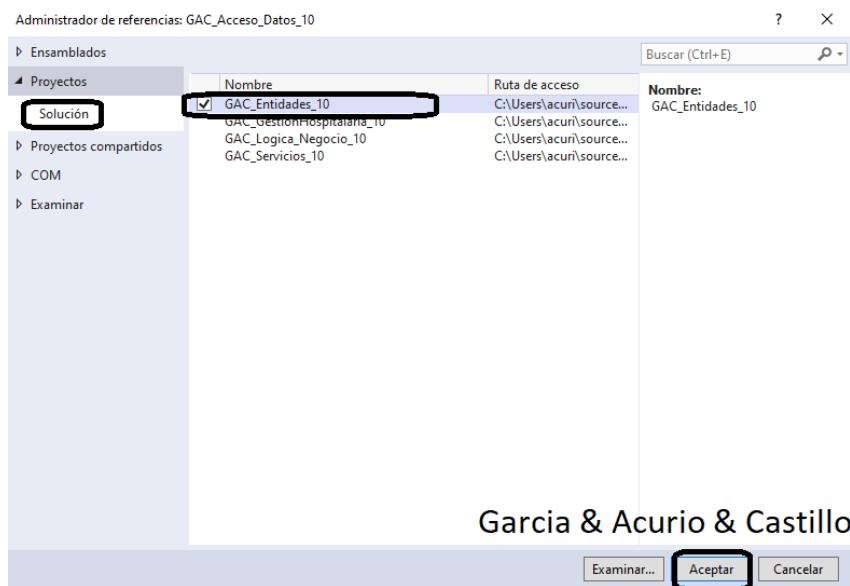
70. Luego una vez agregado tenemos que eliminar un “.tt” que se creó cuando agregamos en el paso anterior con nombre “GAC_GestionHospitalaria_10.Context.tt”



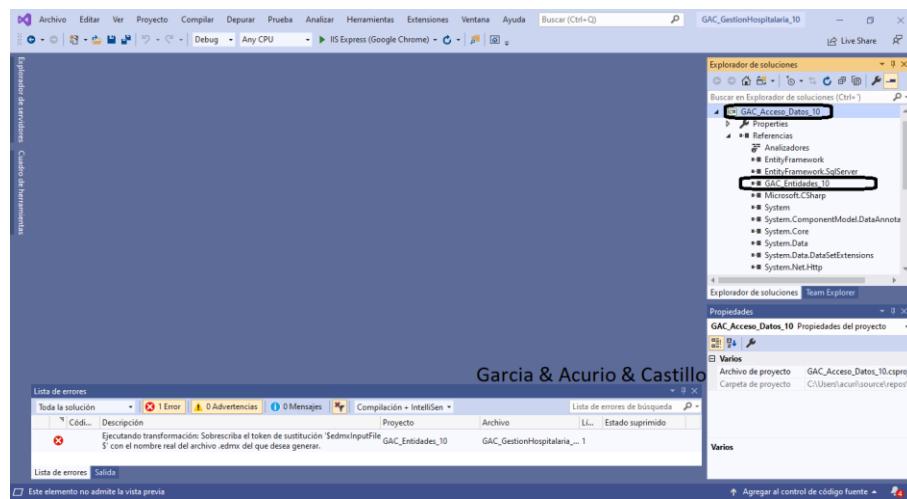
71. Luego una vez eliminado, hacemos clic derecho sobre la “GAC_Acceso_Datos_10” y elegir la opción agregar referencia.



72. Una vez estando en la ventana de referencia, ir a la opción **Solución** y marcar al lado derecho “**GAC_Entidades_10**”, por último, dar en aceptar



73. Verificar que este agregado “**GAC_Entidades_10**”, a las referencias de “**GAC_Acceso_Datos_10**”



74. Despues nos vamos a “**GAC_Entidades_10**”, y luego estando hay abrimos el archivo “**GAC_GestionHospitalaria_10.tt**” y nos dirigimos a la linea 5, borramos todo lo que está entre comillas y agregar el nombre de la Siguiente Capa “**GAC_Acceso_Datos_10**”, seguido del nombre del modelo como se muestra a continuación. Por ultimo, guardamos todo

```

1 <#@ template language="C#" debug="false" hostspecific="true"#
2 <#@ include file="E6.Utility.cs.ttinclude"#"#@#
3 output extension=".cs">#>#<
4
5 const string inputFile = @"..\..\GAC_Acceso_Datos_10\GAC_GestionHospitalaria.edmx";
6 var code = new CodeGenerationTools(this);
7 var ef = new MetadataTools(this);
8 var typeMapper = new TypeMapper(code, ef, textTransform.Errors);
9 var fileManager = EntityFrameworkTemplateFileManager.Create(this);
10 var itemCollection = new EdmMetadataLoader(textTransform.Host, textTransform.Errors).CreateEdmItemCollection(inputFile);
11 var codeStringGenerator = new CodeStringGenerator(code, typeMapper, ef);
12
13 if (!typeMapper.VerifyCaseInsensitiveTypeUniqueness(typeMapper.GetAllGlobalItems(itemCollection), inputFile))
14 {
15     return string.Empty;
16 }
17
18 WriteHeader(codeStringGenerator, fileManager);
19
20 foreach (var entity in typeMapper.GetItemsToGenerate<EntityType>(itemCollection))
21 {
22     fileManager.StartNewFile(entity.Name + ".cs");
23     BeginNamespace(code);
24     #
25     <#=codeStringGenerator.UsingDirectives(inHeader: false)#
26     <#=codeStringGenerator.EntityClassOpening(entity)#
27
28

```

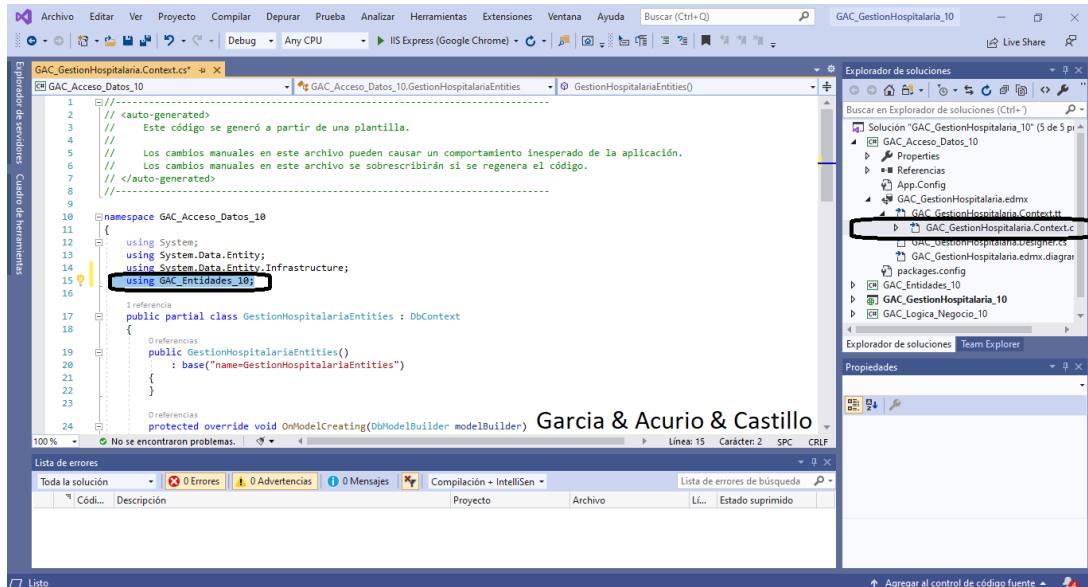
75. Despues nos vamos a la capa “**GAC_Acceso_Datos_10**”, y eliminar un “.tt” con nombre “**GAC_GestionHospitalaria_10.tt**”

```

1 <#@ template language="C#" debug="false" hostspecific="true"#
2 <#@ include file="E6.Utility.cs.ttinclude"#"#@#
3 output extension=".cs">#>#<
4
5 const string inputFile = @"GAC_GestionHospitalaria.edmx";
6 var textTransform = DynamicTextTransformation.Create(this);
7 var code = new CodeGenerationTools(this);
8 var ef = new MetadataTools(this);
9 var typeMapper = new TypeMapper(code, ef, textTransform.Errors);
10 var fileManager = EntityFrameworkTemplateFileManager.Create(this);
11 var itemCollection = new EdmMetadataLoader(textTransform.Host, textTransform.Errors).CreateEdmItemCollection(inputFile);
12 var codeStringGenerator = new CodeStringGenerator(code, typeMapper, ef);
13
14 if (!typeMapper.VerifyCaseInsensitiveTypeUniqueness(typeMapper.GetAllGlobalItems(itemCollection), inputFile))
15 {
16     return string.Empty;
17 }
18
19 WriteHeader(codeStringGenerator, fileManager);
20
21 foreach (var entity in typeMapper.GetItemsToGenerate<EntityType>(itemCollection))
22 {
23     fileManager.StartNewFile(entity.Name + ".cs");
24     BeginNamespace(code);
25     #
26     <#=codeStringGenerator.UsingDirectives(inHeader: false)#
27     <#=codeStringGenerator.EntityClassOpening(entity)#
28

```

76. Nos dirigimos al modelo que se vinculó con la capa “**GAC_Acceso_Datos_10**”, y elegimos la opción “**Context.cs**”, hacer doble clic, y agregar “**using GAC_Entidades_10**”, como otra librería.



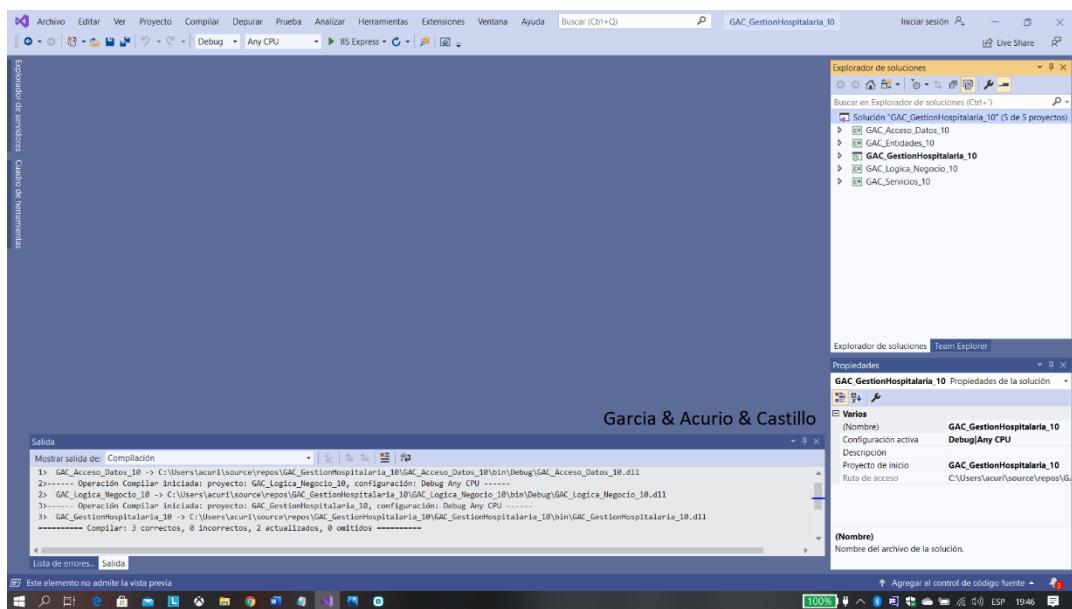
```

1 //<auto-generated>
2 // Este código se generó a partir de una plantilla.
3 //
4 // Los cambios manuales en este archivo pueden causar un comportamiento inesperado de la aplicación.
5 // Los cambios manuales en este archivo se sobreescibirán si se regenera el código.
6 //
7 //</auto-generated>
8
9
10 namespace GAC_Acceso_Datos_10
11 {
12     using System;
13     using System.Data.Entity;
14     using System.Data.Entity.Infrastructure;
15     using GAC_Entidades_10; // Yellow box highlights this line
16
17     [Referencia]
18     public partial class GestionHospitalariaEntities : DbContext
19     {
20         [Referencias]
21         public GestionHospitalariaEntities()
22             : base("name=GestionHospitalariaEntities")
23         {
24         }
25
26         [Referencias]
27         protected override void OnModelCreating(DbModelBuilder modelBuilder)
28             => modelBuilder.Configurations.Add(new GestionHospitalariaConfiguration());
29     }
30 }

```

Garcia & Acurio & Castillo

77. Luego copilamos el proyecto para ver si tenemos errores



```

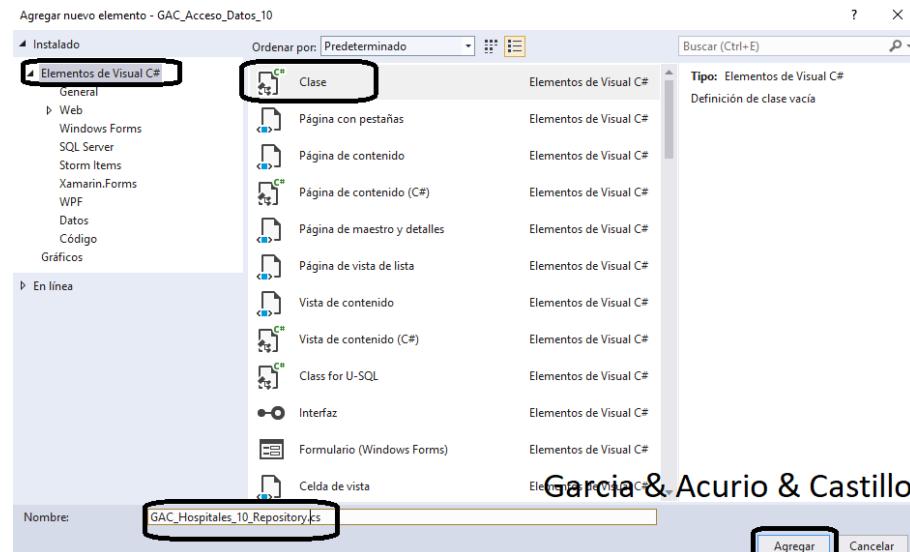
Mostrar salida de: Compilación
1> GAC_Acceso_Datos_10 -> C:\Users\acuris\source\repos\GAC_GestionHospitalaria_10\GAC_Acceso_Datos_10\bin\Debug\GAC_Acceso_Datos_10.dll
2> --> Operación de compilación iniciada: proyecto: GAC_Logica_Negocio_10, configuración: Debug Any CPU .....
3> GAC_Logica_Negocio_10 -> C:\Users\acuris\source\repos\GAC_GestionHospitalaria_10\GAC_Logica_Negocio_10\bin\Debug\GAC_Logica_Negocio_10.dll
4> ..... Operación de compilación iniciada: proyecto: GAC_GestionHospitalaria_10, configuración: Debug Any CPU .....
5> GAC_GestionHospitalaria_10 -> C:\Users\acuris\source\repos\GAC_GestionHospitalaria_10\GAC_GestionHospitalaria_10\bin\GAC_GestionHospitalaria_10.dll
----- Comilar: 3 correctos, 0 incorrectos, 2 actualizados, 0 omitidos -----

```

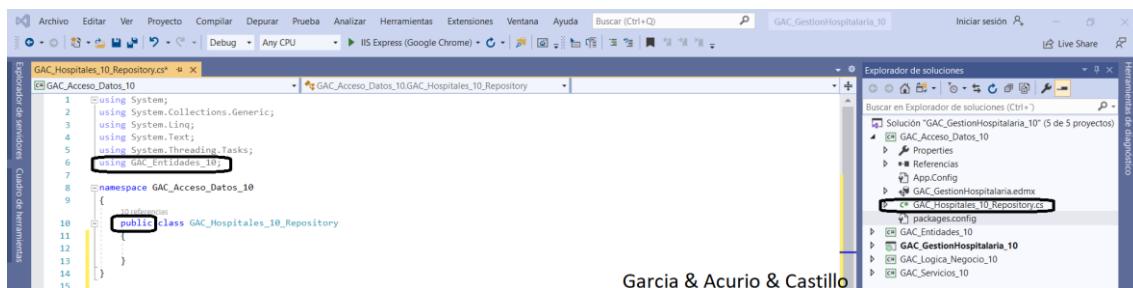
Garcia & Acurio & Castillo

3.9. CREACIÓN DE LOS REPOSITORIOS PARA LA MANIPULACIÓN DE LOS ELEMENTOS DE LA BD

78. Dirigirse a la Capa “**GAC_Acceso_Datos_10**”, clic derecho elegir la opción agregar nuevo, ya hay le damos en clase colocar el nombre “**GAC_Hospitales_10_Repository.cs**”, que va a corresponder a la tabla hospitales.

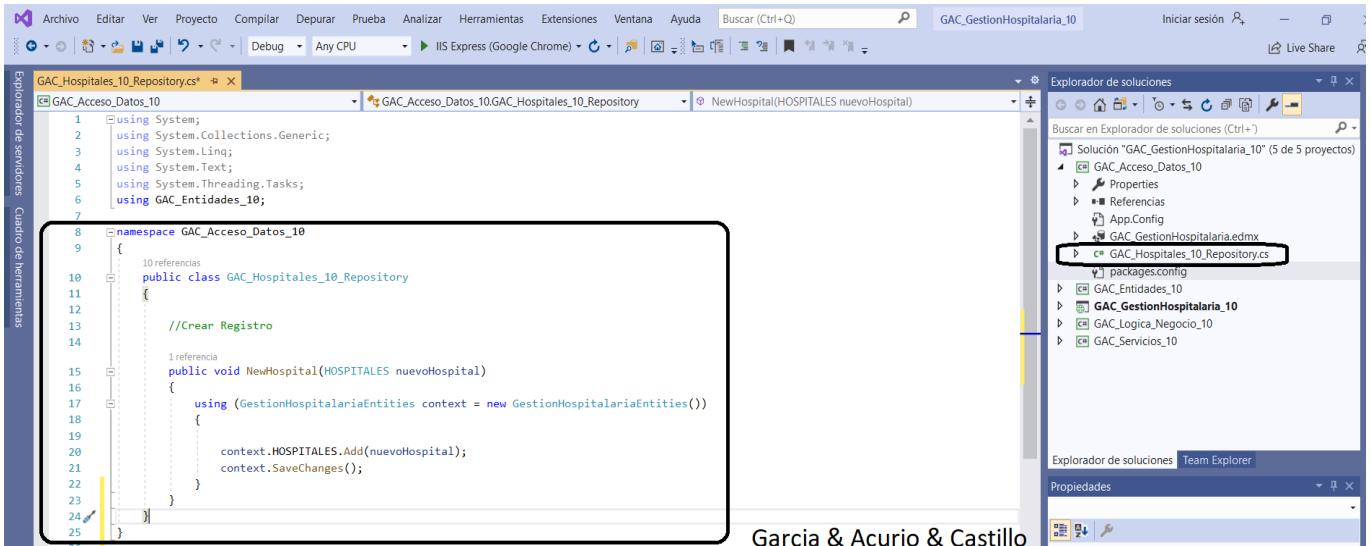


79. Agregar la Capa “**GAC_Entidades_10**”, como biblioteca y colocar la clase como publica como se muestra continuación



3.10. MÉTODOS CRUD DE LA CLASE **“GAC_HOSPITALES_10_REPOSITORY.CS”**

80. Dentro de la clase “**GAC_Hospitales_10_Repository.cs**”, crear el primero método de nombre **NewHospital**.



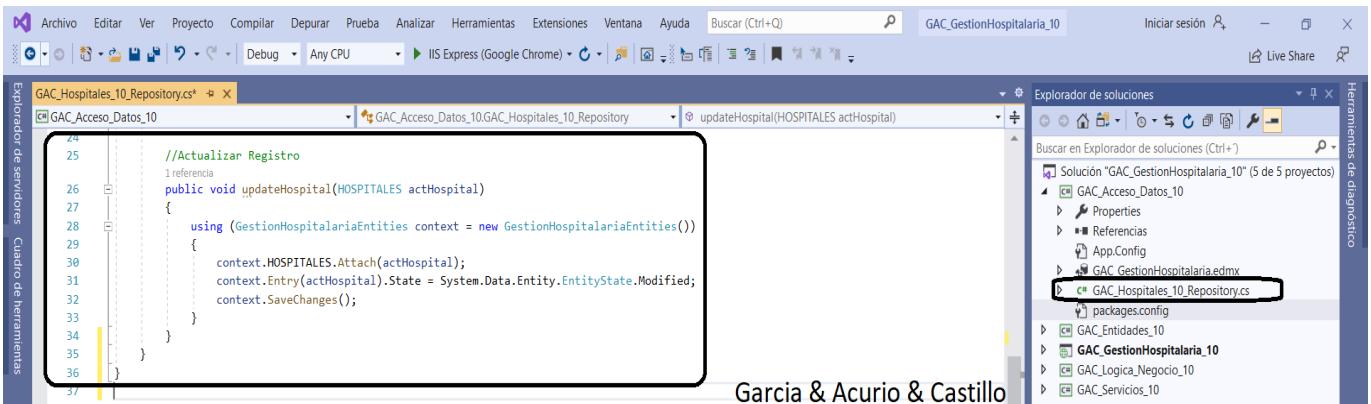
```

1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6  using GAC_Entidades_10;
7
8  namespace GAC_Acceso_Datos_10
9  {
10     public class GAC_Hospitales_10_Repository
11     {
12         //Crear Registro
13
14         public void NewHospital(HOSPITALES nuevoHospital)
15         {
16             using (GestionHospitalariaEntities context = new GestionHospitalariaEntities())
17             {
18                 context.HOSPITALES.Add(nuevoHospital);
19                 context.SaveChanges();
20             }
21         }
22     }
23 }
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50

```

Garcia & Acurio & Castillo

81. Crear el método UpdateHospital



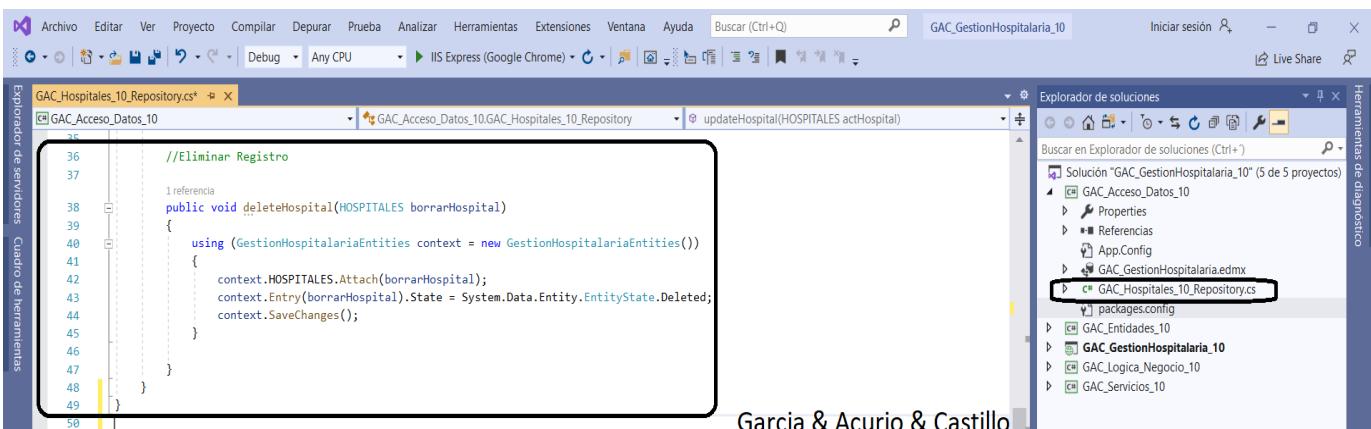
```

24
25
26     //Actualizar Registro
27     public void updateHospital(HOSPITALES actHospital)
28     {
29         using (GestionHospitalariaEntities context = new GestionHospitalariaEntities())
30         {
31             context.HOSPITALES.Attach(actHospital);
32             context.Entry(actHospital).State = System.Data.Entity.EntityState.Modified;
33             context.SaveChanges();
34         }
35     }
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50

```

Garcia & Acurio & Castillo

82. Crear el método DeleteHospital



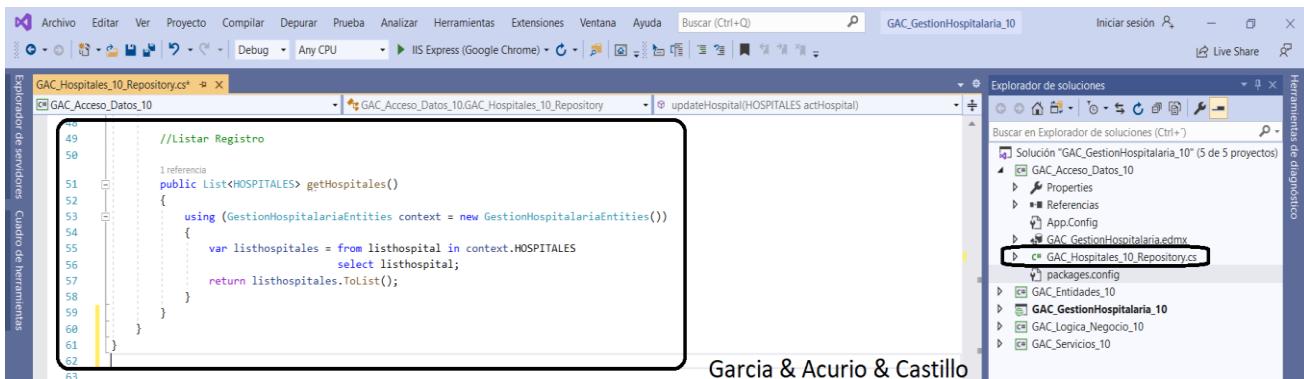
```

36
37
38     //Eliminar Registro
39
40     public void deleteHospital(HOSPITALES borrarHospital)
41     {
42         using (GestionHospitalariaEntities context = new GestionHospitalariaEntities())
43         {
44             context.HOSPITALES.Attach(borrarHospital);
45             context.Entry(borrarHospital).State = System.Data.Entity.EntityState.Deleted;
46             context.SaveChanges();
47         }
48     }
49
50

```

Garcia & Acurio & Castillo

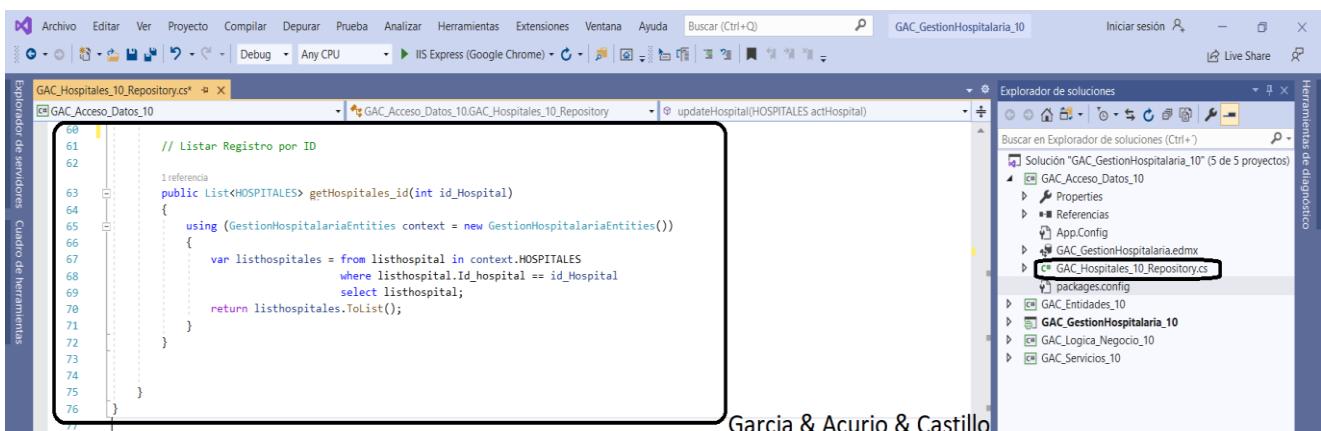
83. Crear método Read, para listar todos los registros



```
49 //Listar Registro
50
51 1 referencia
52 public List<HOSPITALES> getHospitales()
53 {
54     using (GestionHospitalariaEntities context = new GestionHospitalariaEntities())
55     {
56         var listhosptiales = from listhospital in context.HOSPITALES
57             select listhospital;
58         return listhosptiales.ToList();
59     }
60 }
61
62 }
```

Garcia & Acurio & Castillo

84. Crear método Read, para listar por Id

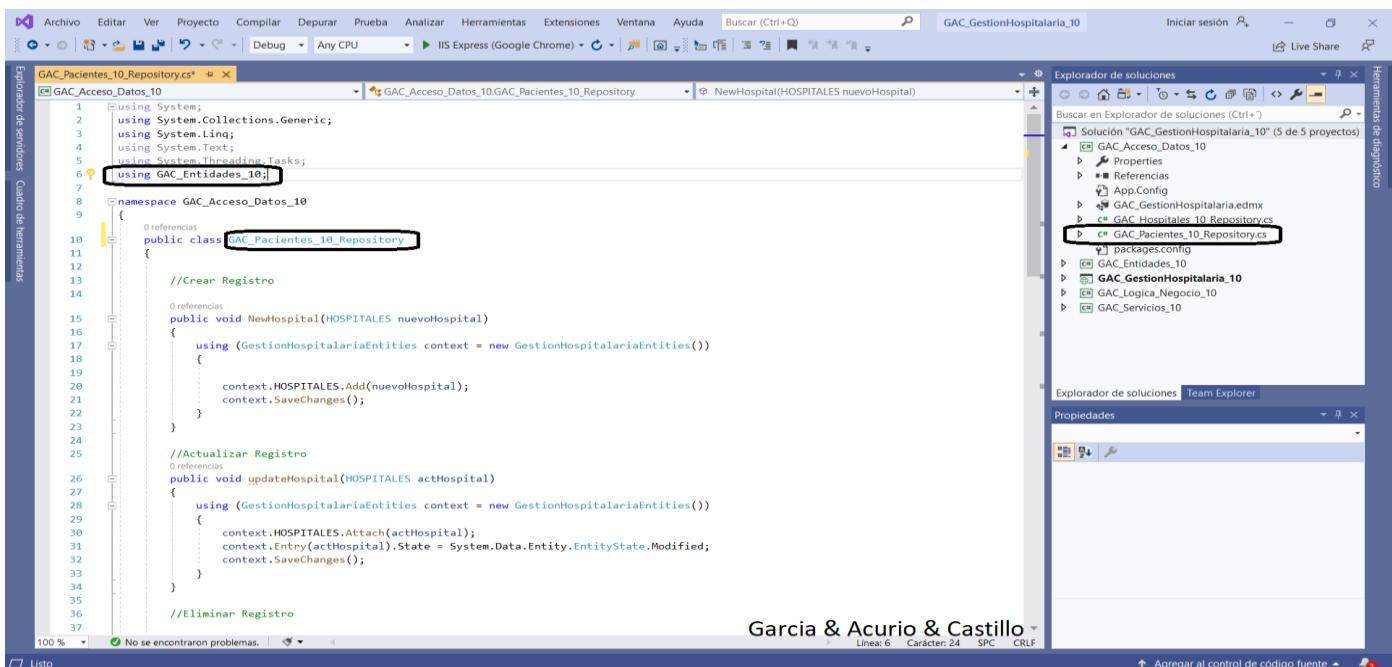


```
60 // Listar Registro por ID
61
62 1 referencia
63 public List<HOSPITALES> getHospitales_id(int id_Hospital)
64 {
65     using (GestionHospitalariaEntities context = new GestionHospitalariaEntities())
66     {
67         var listhosptiales = from listhospital in context.HOSPITALES
68             where listhospital.Id_hospital == id_Hospital
69             select listhospital;
70         return listhosptiales.ToList();
71     }
72 }
73
74
75 }
76 }
```

Garcia & Acurio & Castillo

3.11. CREACIÓN DEL MÉTODO CRUD DE LA CLASE “GAC PACIENTES 10 REPOSITORY.CS”

85. Copiar la clase “GAC_Hospitales_10_Repository.cs”, pegarla en “GAC_Acceso_Datos_10”, y por último le cambiamos el nombre a “GAC_Pacientes_10_Repository.cs”



```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6 using GAC_Entidades_10;
7
8 namespace GAC_Acceso_Datos_10
9 {
10     0 referencias
11     public class GAC_Pacientes_10_Repository
12     {
13
14         //Crear Registro
15
16         0 referencias
17         public void NewHospital(HOSPITALES nuevoHospital)
18         {
19             using (GestionHospitalariaEntities context = new GestionHospitalariaEntities())
20             {
21                 context.HOSPITALES.Add(nuevoHospital);
22                 context.SaveChanges();
23             }
24
25         }
26
27         //Actualizar Registro
28         0 referencias
29         public void updateHospital(HOSPITALES actHospital)
30         {
31             using (GestionHospitalariaEntities context = new GestionHospitalariaEntities())
32             {
33                 context.HOSPITALES.Attach(actHospital);
34                 context.Entry(acthospital).State = System.Data.Entity.EntityState.Modified;
35                 context.SaveChanges();
36             }
37
38         }
39
40         //Eliminar Registro
41
42     }
43 }
44 }
```

Garcia & Acurio & Castillo

86. Luego cambiamos el nombre “HOSPITALES” por “PACIENTES”, para ello hacer clic derecho sobre la palabra “HOSPITALES” y apretamos **ctrl+f**, se nos despliega una venta y le bajamos la flecha, colocaremos a bajo el nombre de la tabla y elegir remplazar todo y así continuamos remplazando todo

```

1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6  using GAC_Entidades_10;
7
8  namespace GAC_Acceso_Datos_10
9  {
10     //referencias
11     public class GAC_Pacientes_10_Repository
12     {
13         //Crear Registro
14         //referencias
15         public void NewHospital(HOSPITALES nuevoHospital)
16         {
17             using (GestionHospitalariaEntities context = new GestionHospitalariaEntities())
18             {
19                 context.HOSPITALES.Add(nuevoHospital);
20                 context.SaveChanges();
21             }
22         }
23
24         //Actualizar Registro
25         //referencias
26         public void updateHospital(HOSPITALES actHospital)
27         {
28             using (GestionHospitalariaEntities context = new GestionHospitalariaEntities())
29             {
30                 context.HOSPITALES.Attach(actHospital);
31                 context.Entry(actHospital).State = System.Data.Entity.EntityState.Modified;
32                 context.SaveChanges();
33             }
34
35         }
36
37         //Eliminar Registro
38     }
39 }

```

87. Luego cambiamos modificamos los **Id_hospital** por **Id_paciente** y continuamos cambiando donde diga hospital o hospitales

```

36
37
38
39     //Eliminar Registro
40
41     public void deletePaciente(PACIENTES borrarPaciente)
42     {
43         using (GestionHospitalariaEntities context = new GestionHospitalariaEntities())
44         {
45             context.PACIENTES.Attach(borrarPaciente);
46             context.Entry(borrarPaciente).State = System.Data.Entity.EntityState.Deleted;
47             context.SaveChanges();
48         }
49
50     }
51
52     //Listar Registro
53
54     public List<PACIENTES> getPacientes()
55     {
56         using (GestionHospitalariaEntities context = new GestionHospitalariaEntities())
57         {
58             var listpacientes = from listpaciente in context.PACIENTES
59             select listpaciente;
60             return listpacientes.ToList();
61         }
62     }
63
64     // Listar Registro por ID
65
66     public List<PACIENTES> getPacientes_id(int id_Paciente)
67     {
68         using (GestionHospitalariaEntities context = new GestionHospitalariaEntities())
69         {
70             var listpacientes = from listpaciente in context.PACIENTES
71             where listpaciente.Id_paciente == id_Paciente
72             select listpaciente;
73             return listpacientes.ToList();
74         }
75     }

```

88. Luego creamos una clase llamada “**Paciente_NombreCompleto.cs**” dentro de la capa “**GAC_Entidades_10**”, y agregamos unos métodos.

The screenshot shows the Visual Studio IDE interface. The top menu bar includes Archivo, Editar, Ver, Proyecto, Compilar, Depurar, Prueba, Analizar, Herramientas, Extensiones, Ventana, Ayuda, Buscar (Ctrl+Q), and IIS Express (Google Chrome). The title bar says "GAC_GestionHospitalaria_10". The left sidebar has "Explorador de servicios" and "Cuadro de herramientas". The main code editor window displays "GAC_Pacientes_10.Repository.cs" with the following code:

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6
7 namespace GAC_Entidades_10
8 {
9     5 referencias
10    public class Paciente_NombreCompleto
11    {
12        1 referencia
13        public int Id_paciente { get; set; }
14
15        1 referencia
16        public string NombreCompleto { get; set; }
17    }
18 }
```

The "Explorador de soluciones" (Solution Explorer) on the right shows the solution structure:

- Solución "GAC_GestionHospitalaria_10" (5 de 5 proyectos)
 - GAC_Acceso_Datos_10
 - GAC_Entidades_10 (selected)
 - GAC_GestionHospitalaria_10.tt
 - GAC_Logica_Negocio_10
 - GAC_Servicios_10

A yellow box highlights the "Paciente_NombreCompleto.cs" file under "GAC_Entidades_10".

89. Después de haber creado eso creamos el método para listar por nombre completo

The screenshot shows the Visual Studio IDE interface. The top menu bar includes Archivo, Editar, Ver, Proyecto, Compilar, Depurar, Prueba, Analizar, Herramientas, Extensiones, Ventana, Ayuda, Buscar (Ctrl+Q), and IIS Express (Google Chrome). The title bar says "GAC_GestionHospitalaria_10". The left sidebar has "Explorador de servicios" and "Cuadro de herramientas". The main code editor window displays "GAC_Pacientes_10.Repository.cs" with the following code:

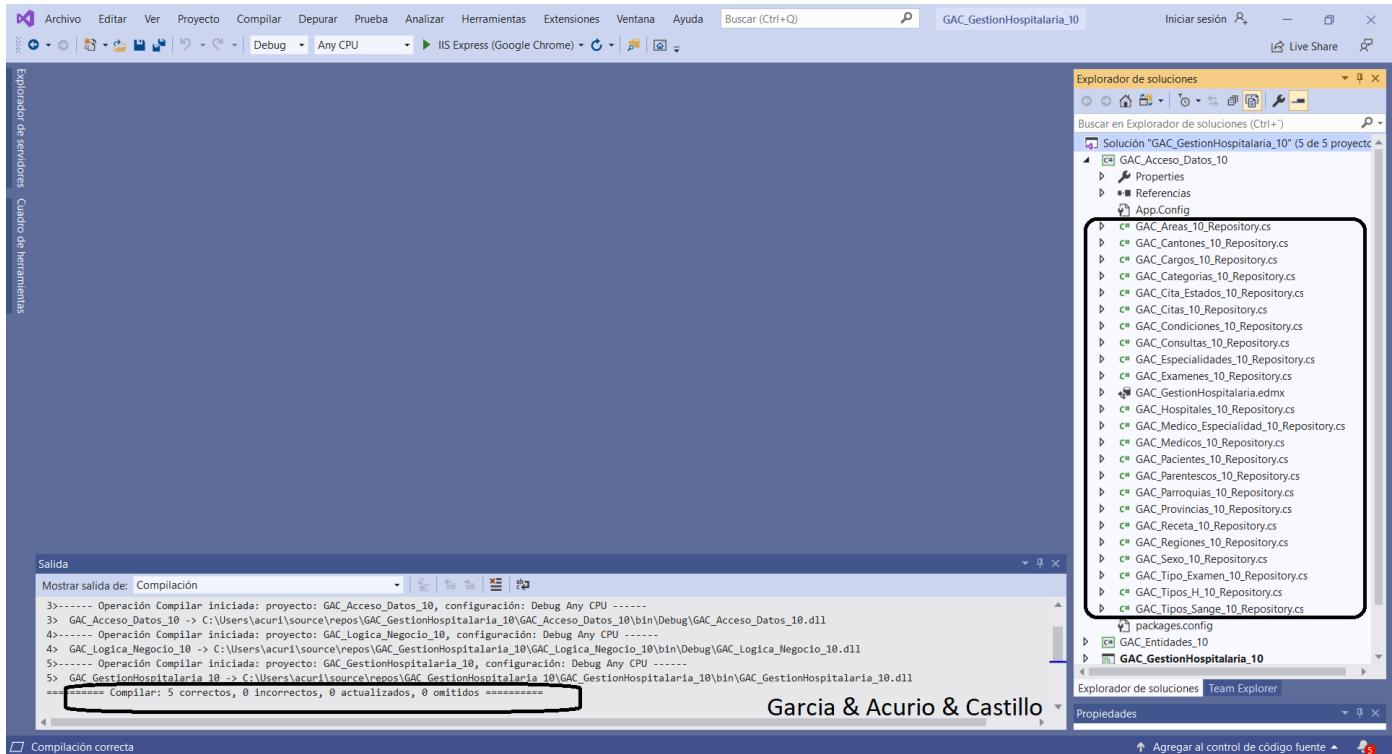
```
67     var listpacientes = from listpaciente in context.PACIENTES
68         where listpaciente.Id_paciente == id_Paciente
69         select listpaciente;
70     return listpacientes.ToList();
71 }
72
73 //Listar Registro Nombre Completo
74
75 public List<Paciente_NombreCompleto> getPacientes_NombreCompleto()
76 {
77     using (GestionHospitalariaEntities context = new GestionHospitalariaEntities())
78     {
79         var listpacientes = from listpaciente in context.PACIENTES
80
81             select new Paciente_NombreCompleto{Id_paciente=listpaciente.Id_paciente,NombreCompleto=listpaciente.Apellido};
82
83         return listpacientes.ToList();
84     }
85
86 }
87
88 }
```

The "Explorador de soluciones" (Solution Explorer) on the right shows the solution structure:

- Solución "GAC_GestionHospitalaria_10" (5 de 5 proyectos)
 - GAC_Acceso_Datos_10
 - GAC_Entidades_10
 - GAC_GestionHospitalaria_10 (selected)
 - GAC_Logica_Negocio_10
 - GAC_Servicios_10

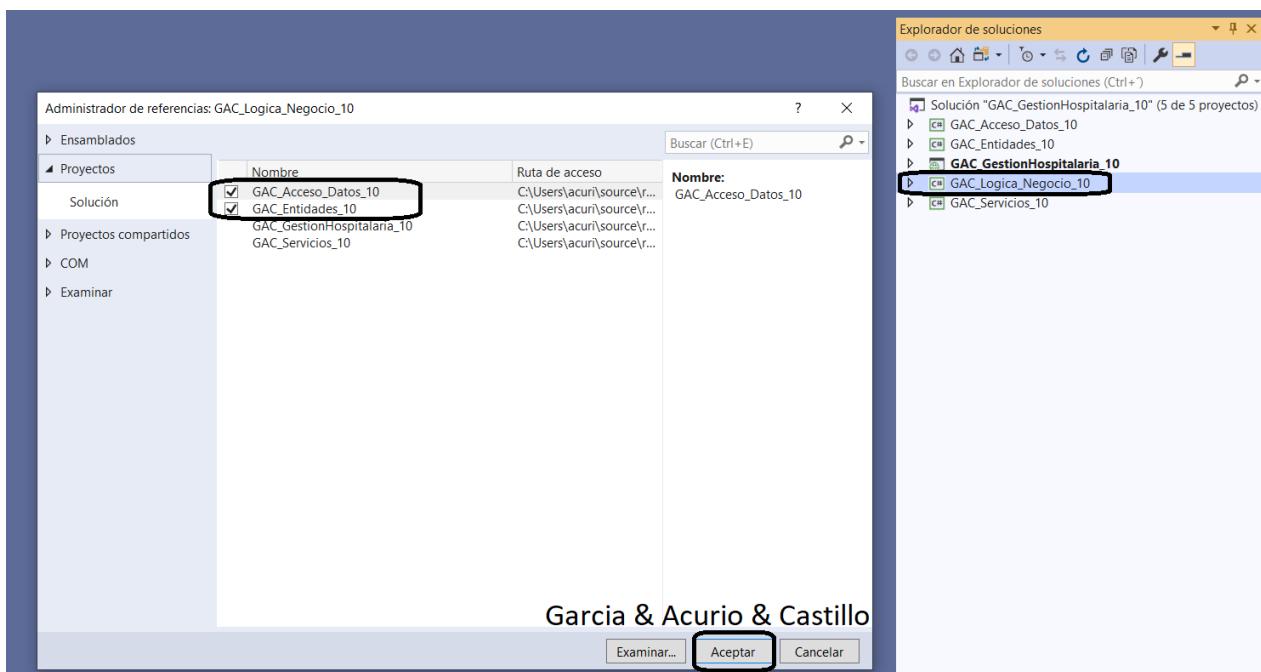
A yellow box highlights the "GAC_Pacientes_10.Repository.cs" file under "GAC_GestionHospitalaria_10".

90. Luego hacemos el mismo proceso para la creación de las **21 clases restantes** que corresponde a las tablas de la base de datos, en la siguiente imagen podemos verificar todas clases creadas, copilamos el proyecto para verificar si está en buen funcionamiento

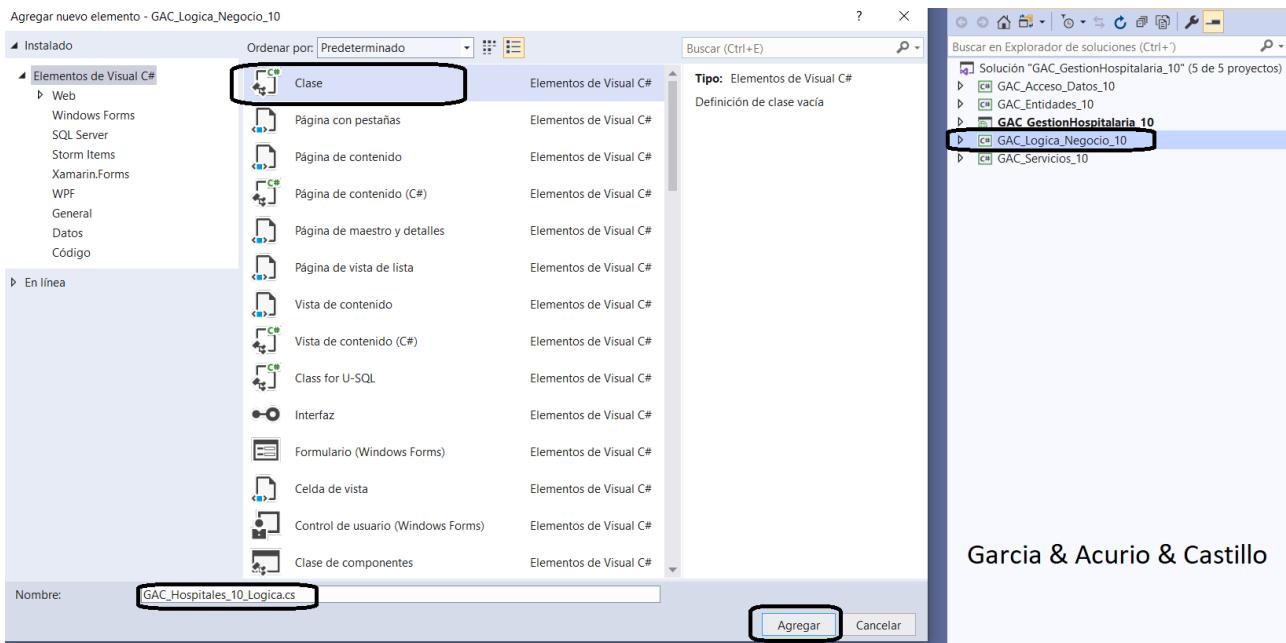


3.12. CREACIÓN DE CLASES Y MÉTODOS EN LA CAPA “GAC_LOGICA_NEGOCIO_10”

91. Creamos una referencia dentro de la Capa “**GAC_Logica_Negocio_10**”, con la Capa “**GAC_Acceso_Datos_10**” y la Capa “**GAC_Entidades_10**” y le damos en Aceptar

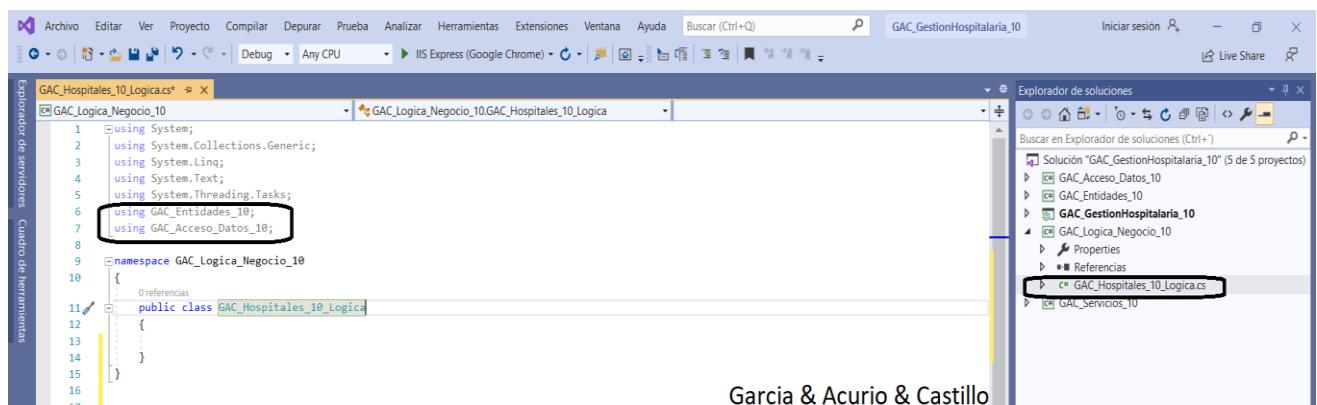


92. Luego creamos una Clase de nombre “**GAC_Hospitales_10_Logica**” en La Capa “**GAC_Logica_Negocio_10**”



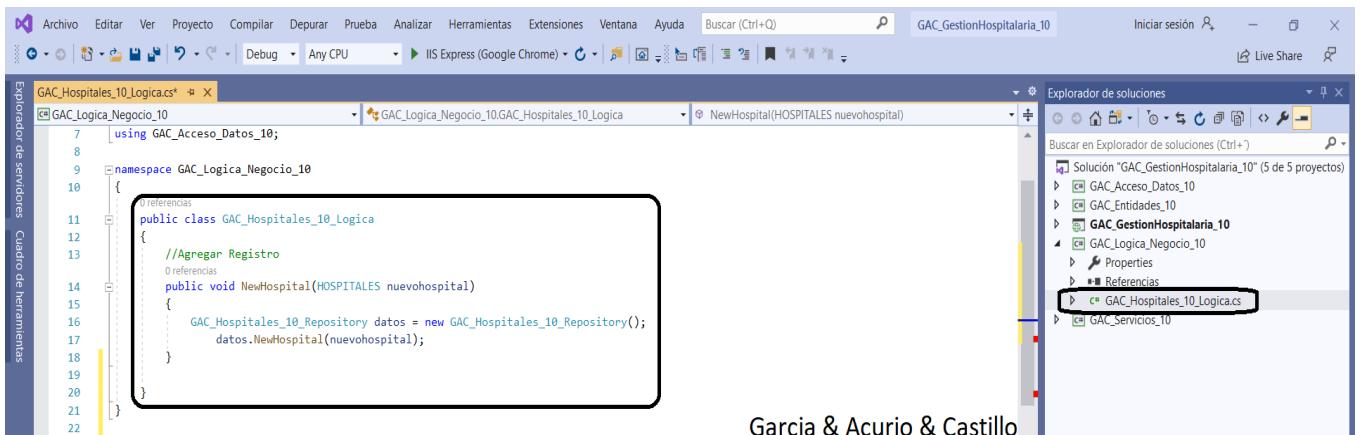
Garcia & Acurio & Castillo

93. Luego al abrimos la clase creada y agregamos la capa “**GAC_Entidades_10**” y “**GAC_Acceso_Datos_10**” como biblioteca de Using



Garcia & Acurio & Castillo

94. Creamos el Método **NewHospital**.



Garcia & Acurio & Castillo

95. Creamos el Método UpdateHospital

The screenshot shows the Visual Studio IDE with the solution "GAC_GestionHospitalaria_10" open. The code editor displays the file "GAC_Hospitales_10_Logica.cs" containing the following C# code:

```
//Actualizar Registro
public void updateHospital(HOSPITALES actHospital)
{
    GAC_Hospitales_10_Repository datos = new GAC_Hospitales_10_Repository();
    datos.updateHospital(actHospital);
}
```

The code is highlighted with a black rectangle. The Solution Explorer on the right shows the project structure with the file "GAC_Hospitales_10_Logica.cs" selected.

Garcia & Acurio & Castillo

96. Creamos el Método DeleteHospital

The screenshot shows the Visual Studio IDE with the solution "GAC_GestionHospitalaria_10" open. The code editor displays the file "GAC_Hospitales_10_Logica.cs" containing the following C# code:

```
//Eliminar Registro
public void deleteHospital(HOSPITALES borrarHospital)
{
    GAC_Hospitales_10_Repository datos = new GAC_Hospitales_10_Repository();
    datos.deleteHospital(borrarHospital);
}
```

The code is highlighted with a black rectangle. The Solution Explorer on the right shows the project structure with the file "GAC_Hospitales_10_Logica.cs" selected.

Garcia & Acurio & Castillo

97. Creamos el Método Read, para listar todos los registros

The screenshot shows the Visual Studio IDE with the solution "GAC_GestionHospitalaria_10" open. The code editor displays the file "GAC_Hospitales_10_Logica.cs" containing the following C# code:

```
//listar todos los Registros
public List<HOSPITALES> getHospitales()
{
    List<HOSPITALES> listhosptiales = new List<HOSPITALES>();
    GAC_Hospitales_10_Repository datos = new GAC_Hospitales_10_Repository();
    listhosptiales = datos.getHospitales();
    return listhosptiales;
}
```

The code is highlighted with a black rectangle. The Solution Explorer on the right shows the project structure with the file "GAC_Hospitales_10_Logica.cs" selected.

Garcia & Acurio & Castillo

98. Creamos el Método Read, para listar por Id

The screenshot shows the Visual Studio IDE with the solution "GAC_GestionHospitalaria_10" open. The code editor displays the file "GAC_Hospitales_10_Logica.cs" containing the following C# code:

```
//listar Registro por Id
public List<HOSPITALES> getHospitales_id(int id_Hospital)
{
    List<HOSPITALES> listhosptiales = new List<HOSPITALES>();
    GAC_Hospitales_10_Repository datos = new GAC_Hospitales_10_Repository();
    listhosptiales = datos.getHospitales_id(id_Hospital);
    return listhosptiales;
}
```

The code is highlighted with a black rectangle. The Solution Explorer on the right shows the project structure with the file "GAC_Hospitales_10_Logica.cs" selected.

Garcia & Acurio & Castillo

99. Copiar la Clase “**GAC_Hospitales_10_Logica**” y pegarla en “**GAC_Logica_Negocio_10**” y cambiarle a la clase con este nombre “**GAC_Pacientes_10_Logica**”

```

1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6  using GAC_Entidades_10;
7  using GAC_Acceso_Datos_10;
8
9  namespace GAC_Logica_Negocio_10
10 {
11     public class GAC_Pacientes_10_Logica
12     {
13         //Agregar Registro
14         public void NewHospital(HOSPITALES nuevohospital)
15         {
16             GAC_Hospitales_10_Repository datos = new GAC_Hospitales_10_Repository();
17             datos.NewHospital(nuevohospital);
18         }
19     }
}

```

Garcia & Acurio & Castillo

100. Luego reemplazamos en todos los métodos de la clase copiada el nombre **Hospitales** por **Pacientes** y agregamos el método de listar por **Nombre Completo**

```

9  namespace GAC_Logica_Negocio_10
10 {
11     public class GAC_Pacientes_10_Logica
12     {
13         //Agregar Registro
14         public void NewPacientes(PACIENTES nuevoPaciente)
15         {
16             GAC_Pacientes_10_Repository datos = new GAC_Pacientes_10_Repository();
17             datos.NewPaciente(nuevoPaciente);
18         }
19
20         //Actualizar Registro
21
22         public void updatePaciente(PACIENTES actPaciente)
23         {
24             GAC_Pacientes_10_Repository datos = new GAC_Pacientes_10_Repository();
25             datos.updatePaciente(actPaciente);
26         }
27
28         //Eliminar Registro
29
30         public void deletePaciente(PACIENTES borrarpaciente)
31         {
32             GAC_Pacientes_10_Repository datos = new GAC_Pacientes_10_Repository();
33             datos.deletePaciente(borrarpaciente);
34         }
35
36     }
37 }

```

Garcia & Acurio & Castillo

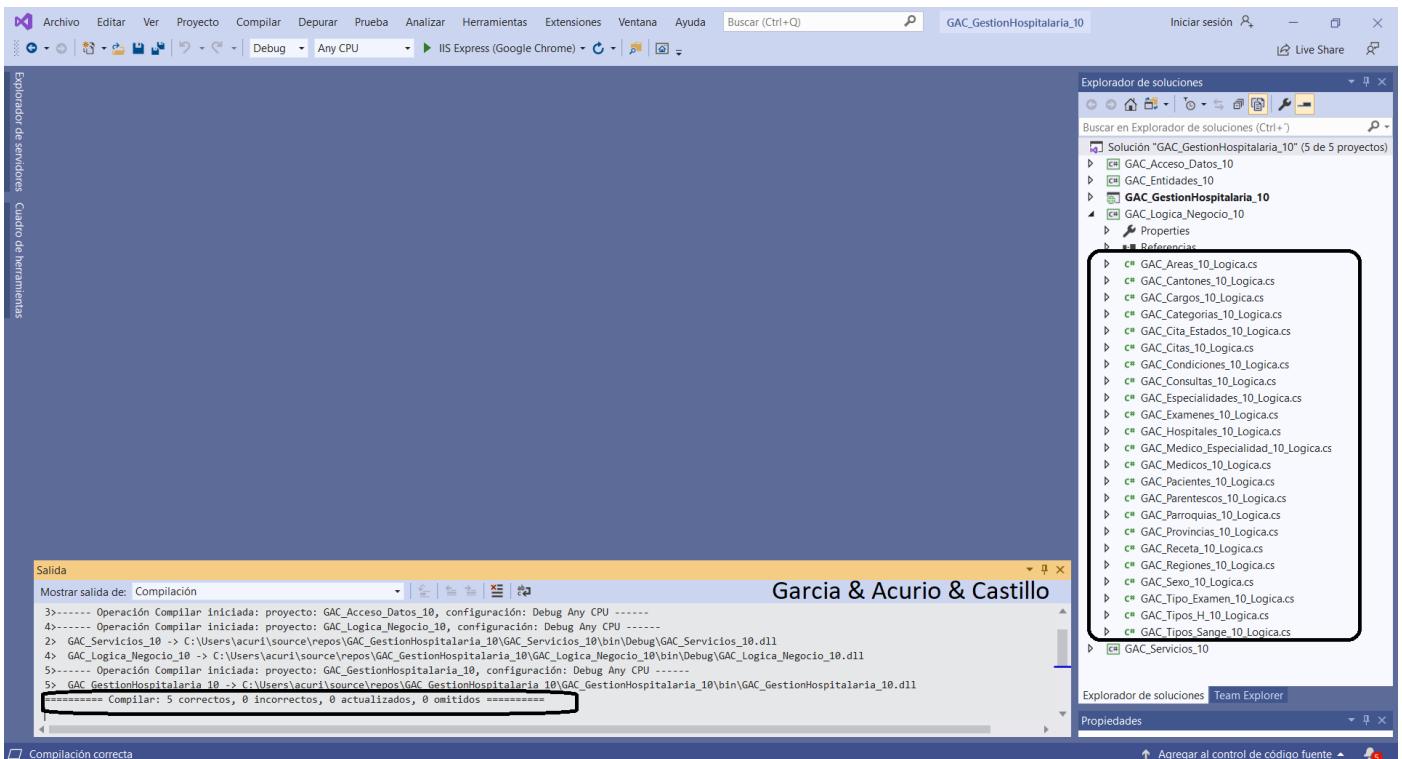
```

37
38     //listar todos los Registros
39     public List<PACIENTES> getPacientes()
40     {
41         List<PACIENTES> listPacientes = new List<PACIENTES>();
42         GAC_Pacientes_10_Repository datos = new GAC_Pacientes_10_Repository();
43         listPacientes = datos.getPacientes();
44         return listPacientes;
45     }
46
47     //listar Registro por Id
48     public List<PACIENTES> getPacientes_id(int id_Paciente)
49     {
50
51         List<PACIENTES> listPacientes = new List<PACIENTES>();
52         GAC_Pacientes_10_Repository datos = new GAC_Pacientes_10_Repository();
53         listPacientes = datos.getPacientes_id(id_Paciente);
54         return listPacientes;
55     }
56
57     // listar Nombre Completo
58     public List<Paciente_NombreCompleto> getPacientes_NombreCompleto()
59     {
60
61         List<Paciente_NombreCompleto> listPacientes = new List<Paciente_NombreCompleto>();
62         GAC_Pacientes_10_Repository datos = new GAC_Pacientes_10_Repository();
63         listPacientes = datos.getPacientes_NombreCompleto();
64         return listPacientes;
65     }
66
67 }

```

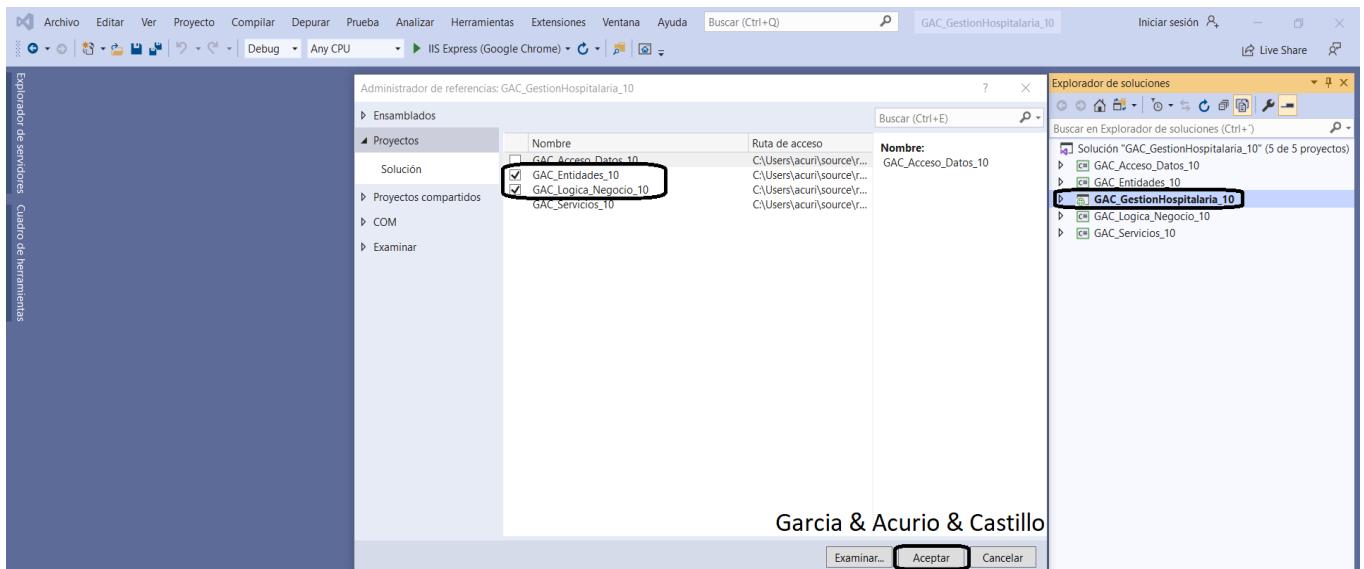
Garcia & Acurio & Castillo

101. Luego hacemos los 2 pasos anteriores para la creación de las **21 clases** restantes como lo muestra en la imagen, corremos y verificamos q no tenemos errores

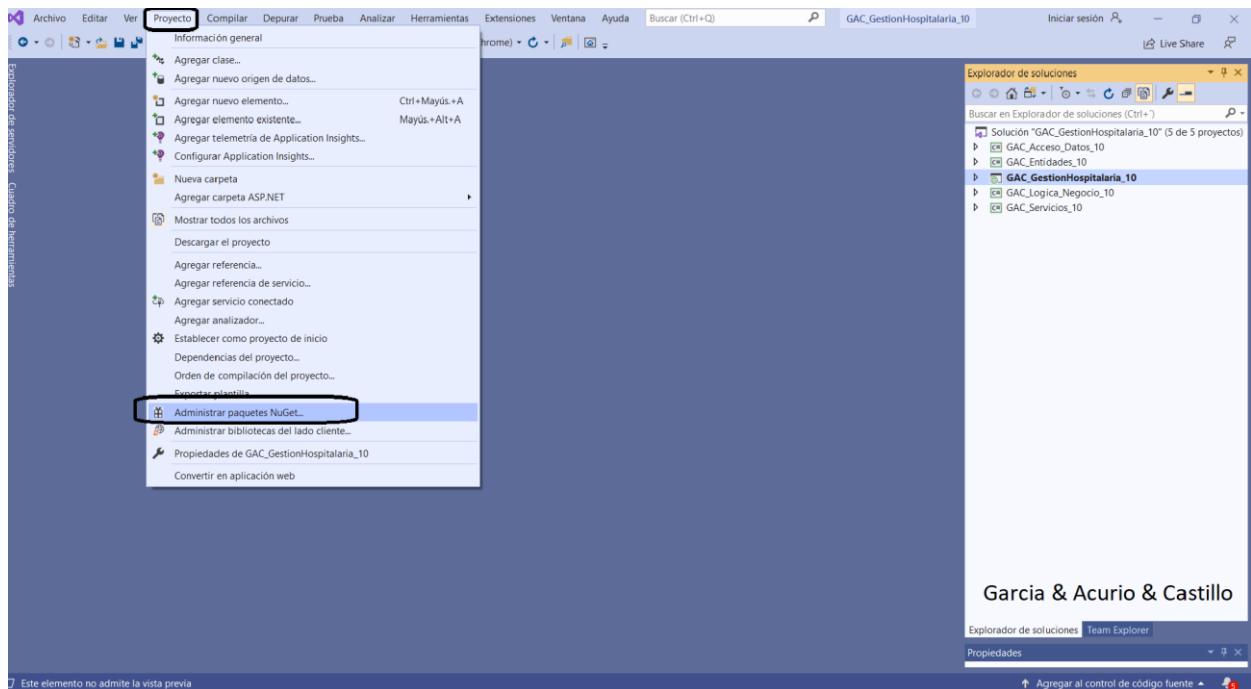


3.13. PROGRAMACIÓN DE LA CAPA DE PRESENTACIÓN

102. Principalmente nos dirigimos a la Capa Principal “**GAC_GestionHospitalaria_10**”, damos Clic derecho sobre esa solución, **agregar-referencia**, elegimos las capas “**GAC_Entidades_10**” y “**GAC_Logica_Negocio_10**” y aceptamos

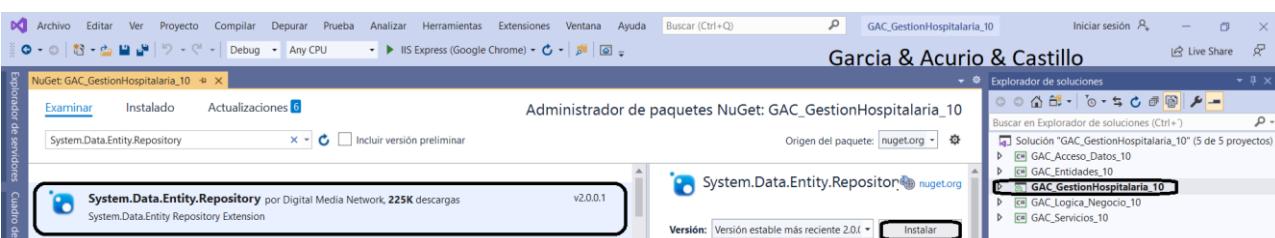
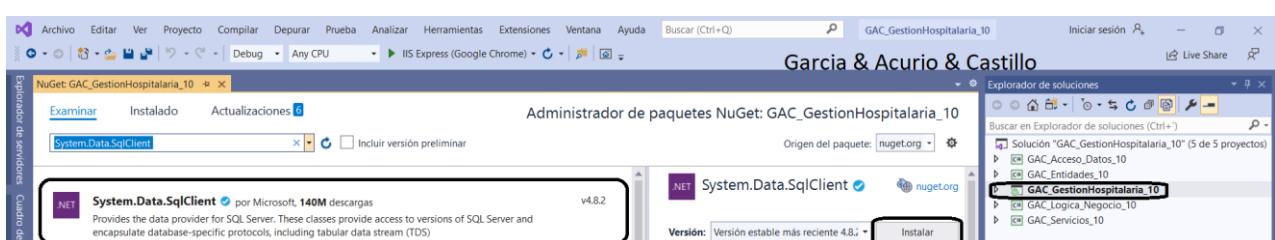
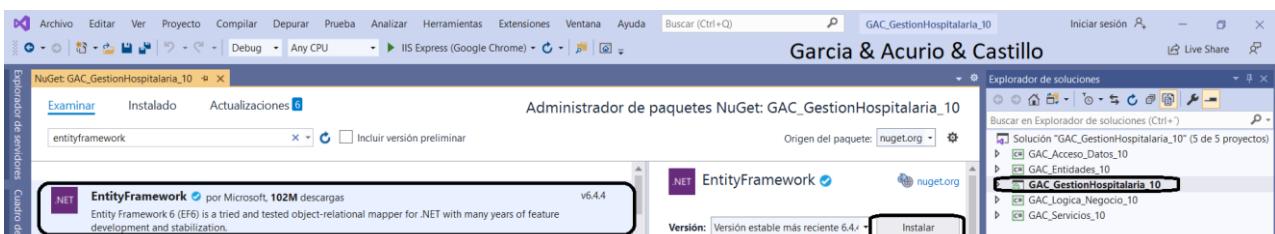


103. Luego nos dirigimos a **proyecto-administrar paquetes NuGet**

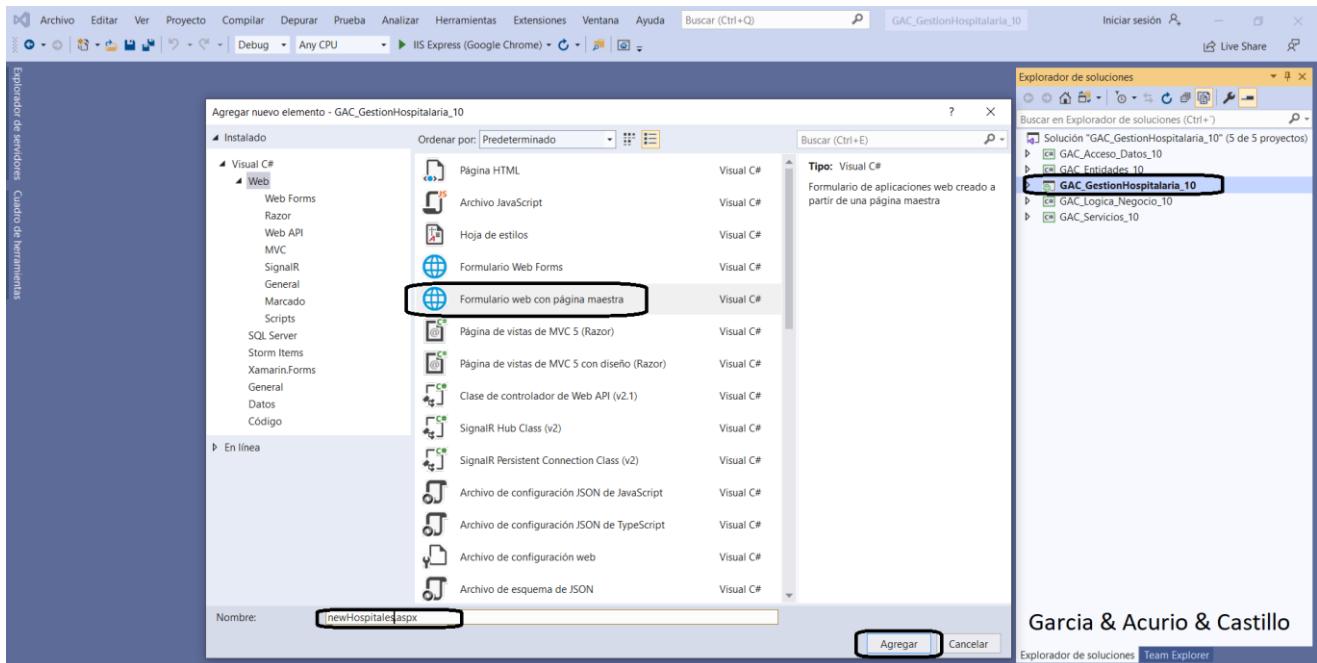


104. Una vez adentro buscamos y descargamos los siguientes componentes

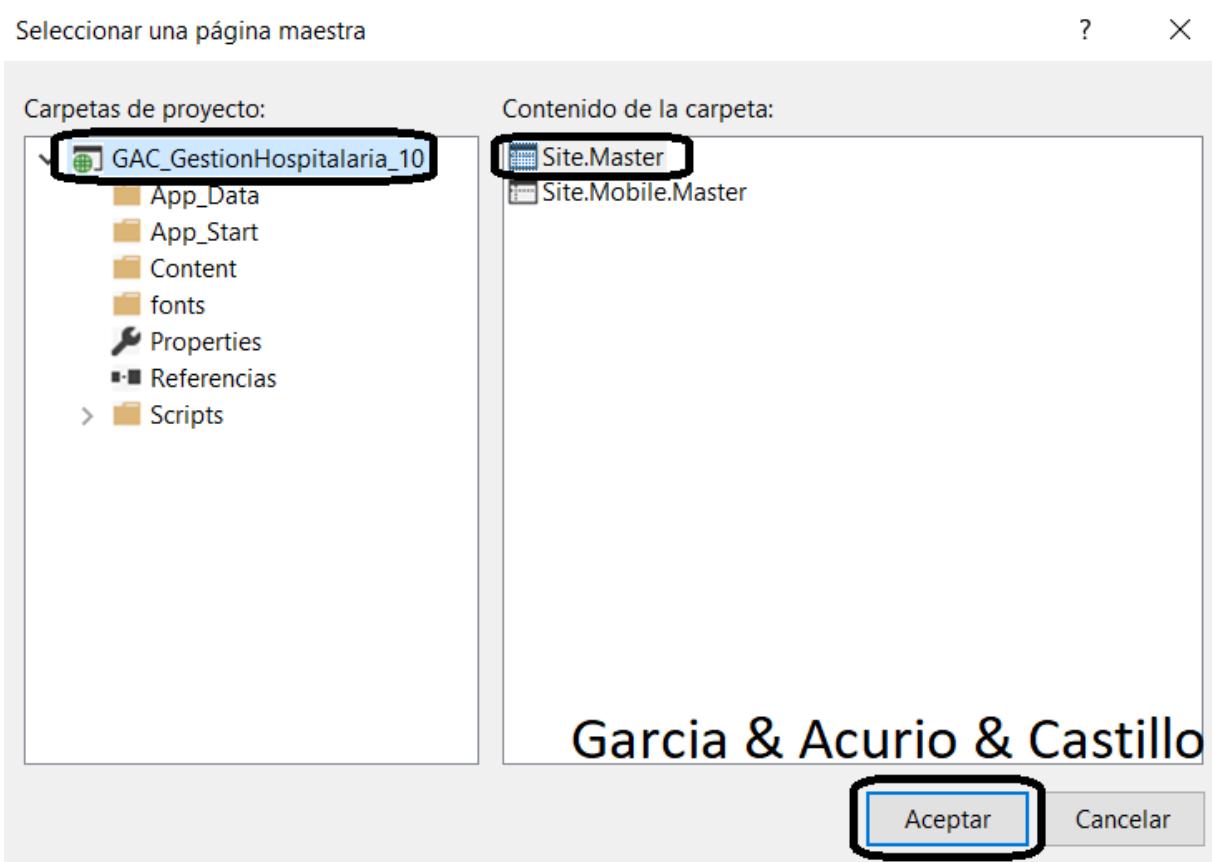
- **EnttiyFrameWork**
- **System.Data.SqlClient**
- **System.Data.Entity.Repository**



105. Luego hacer clic derecho sobre la capa de presentación y **agregar- nuevo elemento**, elegimos formulario de tipo web con la opción de página maestra, colocamos el nombre de **newHospitales**



106. Se nos mostrara una ventana, elegimos la opción **Site.Master** y luego aceptamos

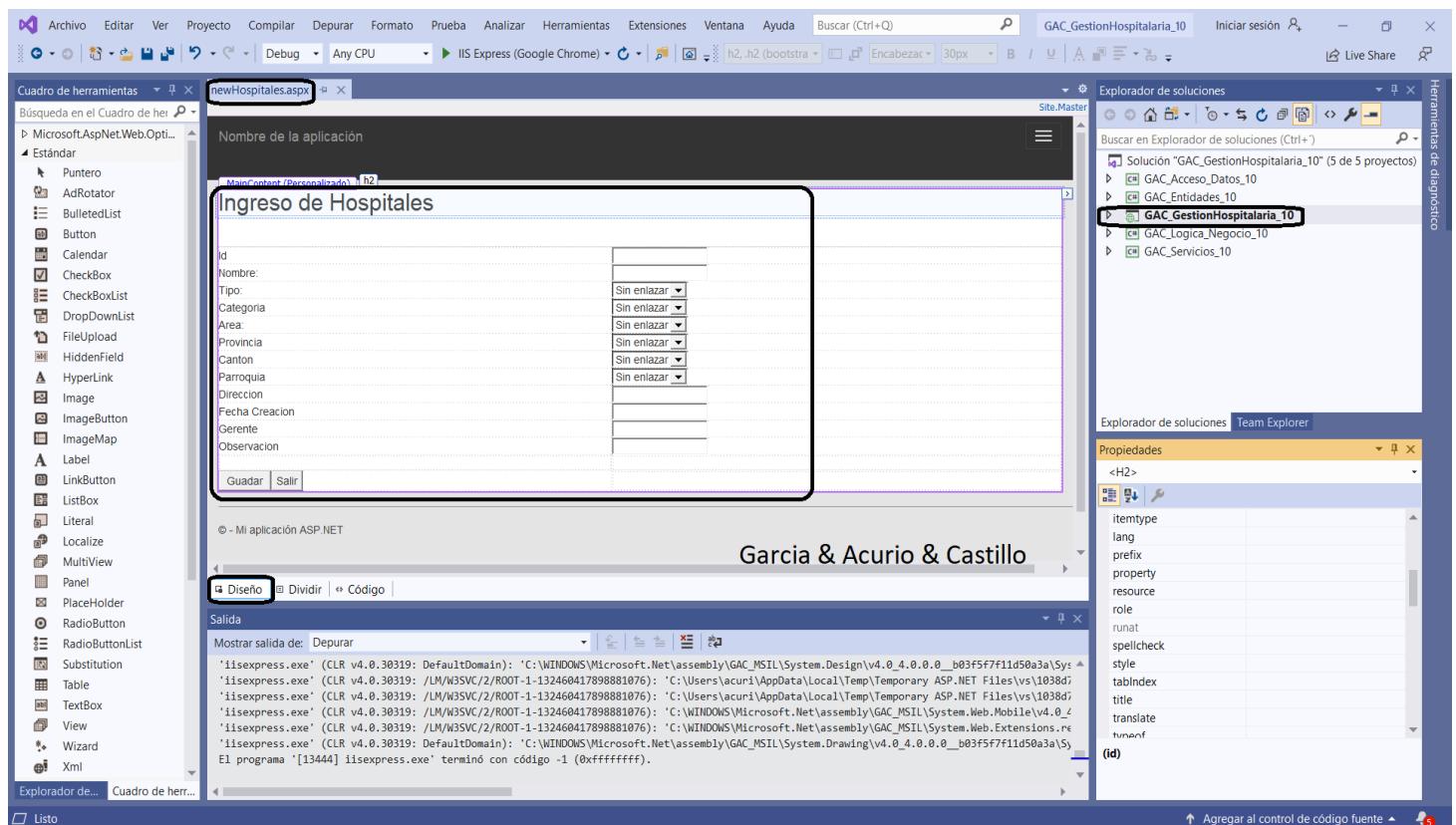


107. Una vez aceptado se nos abrirá archivo que se creó, creamos una tabla con 2 columnas y 13 fila que corresponde a los campos de las tablas **Hospital** y la última corresponde a los **botones**. Le pondremos el nombre **Hospitales**

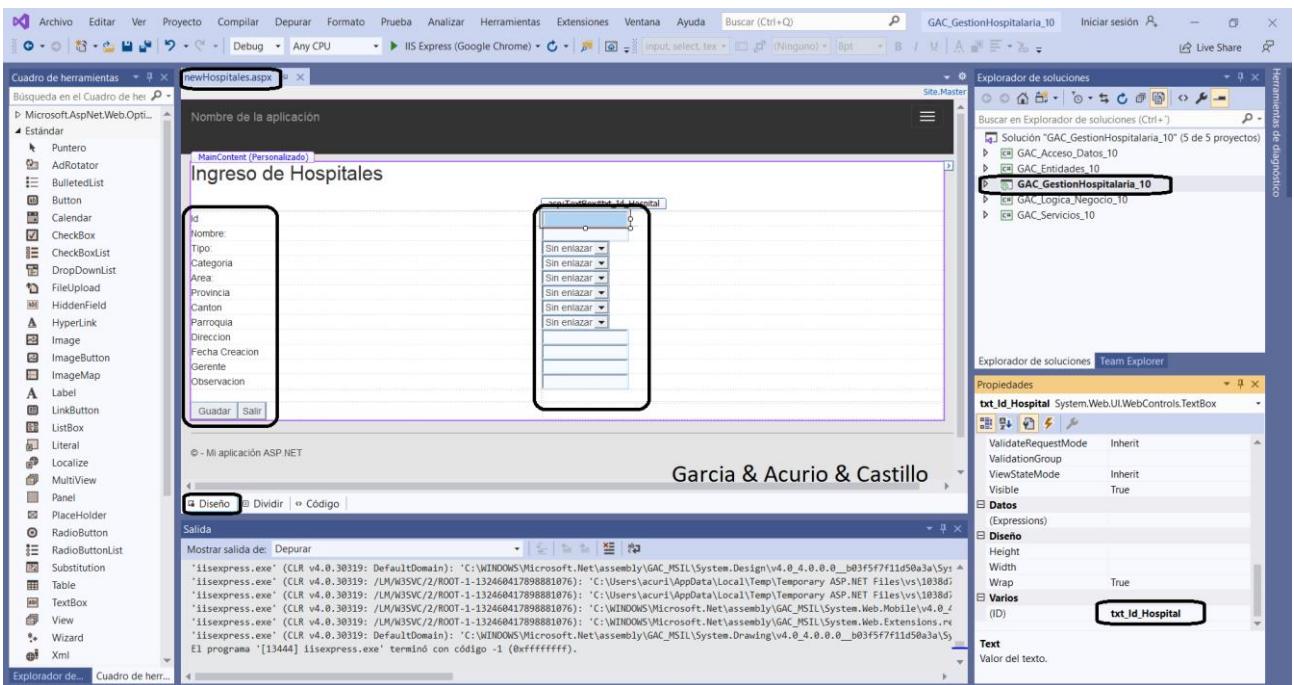


Garcia & Acurio & Castillo

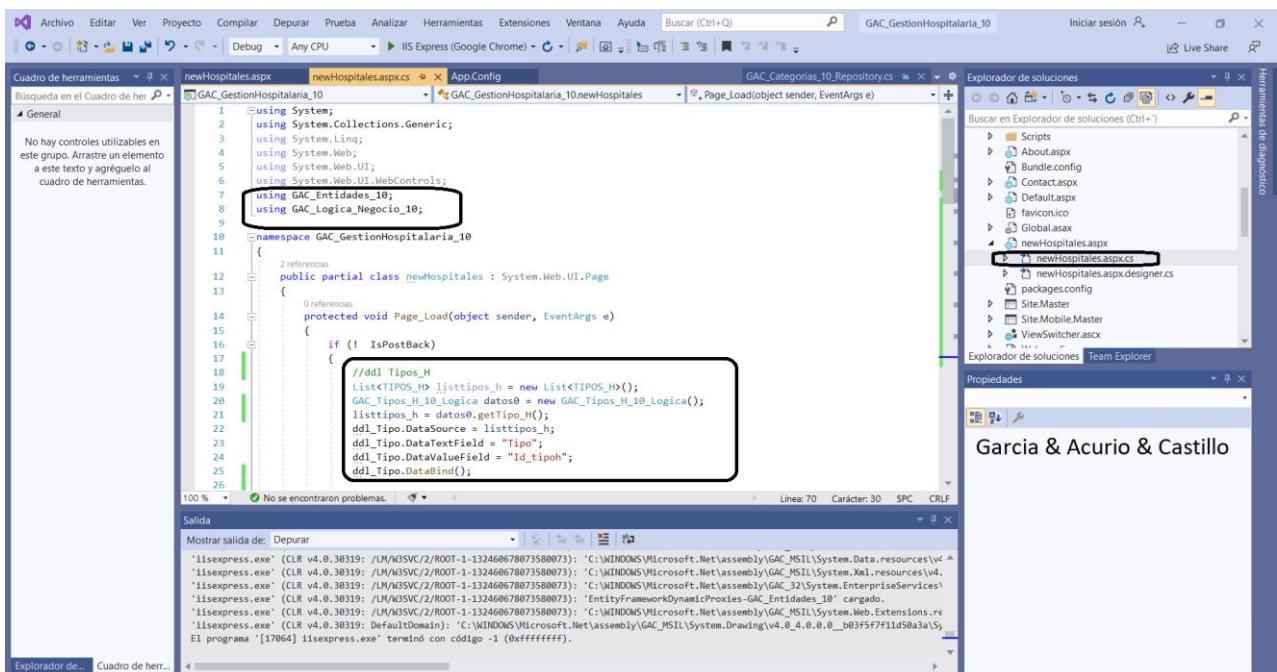
108. Luego creamos los controles, hacemos clic sobre la opción **Diseño** en la parte inferior **izquierda**, y en las filas colocar **12 Label**, **6 TextBox**, **6 DropDownList** y colocamos los nombres de los campos correspondientes y en la última fila colocar **2 Button** guardar y salir



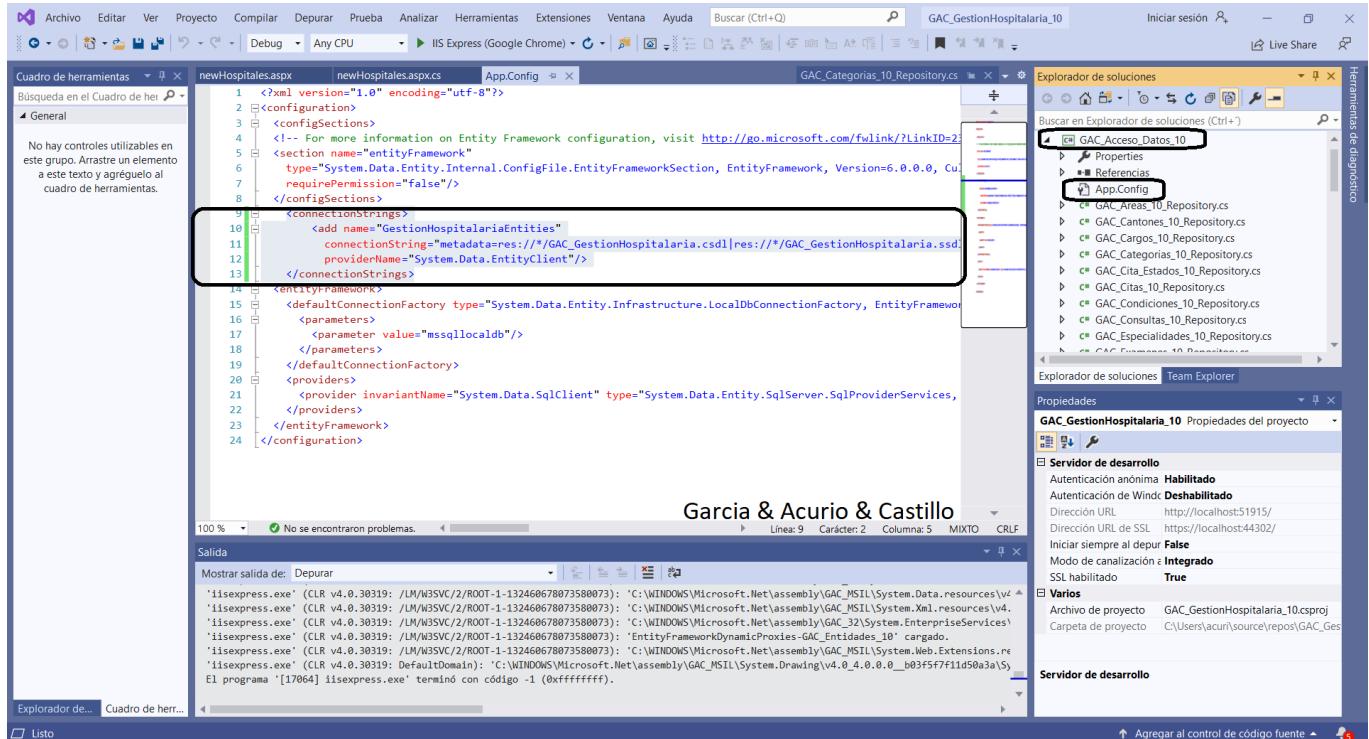
109. Colocarles nombre a los controles para poder identificar mejor



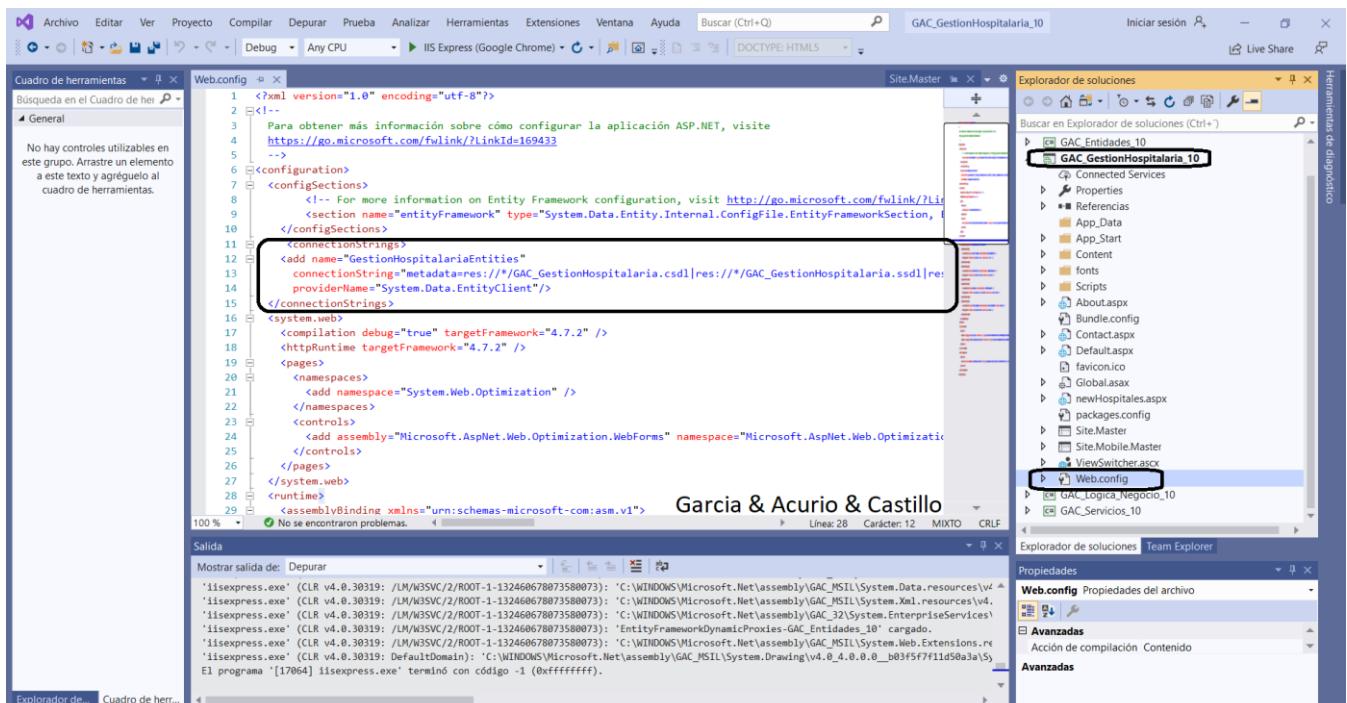
110. Entramos a clases **newHospitales.aspx.cs**, y agregar como biblioteca la Capa **“GAC_Entidades_10”** y la Capa **“GAC_Logica_Negocio_10”**, luego agregamos el siguiente código para **Tipo_H**



111. Nos dirigimos a la Capa “**GAC_Acceso_Datos_10**”, entramos en **App.Config** y copiamos desde la línea 9 a la 13, que es el código que se genera cuando se conecta la aplicación con SQL.



112. Luego pegamos las líneas de código copiadas en el paso anterior dentro de la Capa “**GAC_GestionHospitalaria_10**”, en **Web.Config** después de la línea 10



113. Luego entramos a **newHospitales.aspx.cs**, copiamos y editamos **tipo_h** varias veces con los de mas atributos de la tabla

```

26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61

```

```

//ddl Categorias
List<CATEGORIAS> listcategorias = new List<CATEGORIAS>();
GAC_Categorias_10_Logica datos1 = new GAC_Categorias_10_Logica();
listcategorias = datos1.getCategories();
ddl_Categorias.DataSource = listcategorias;
ddl_Categorias.DataTextField = "Categoria";
ddl_Categorias.DataValueField = "Id_categoria";
ddl_Categorias.DataBind();

//ddl Areas
List<AREAS> listareas = new List<AREAS>();
GAC_Areas_10_Logica datos2 = new GAC_Areas_10_Logica();
listareas = datos2.getAreas();
ddl_area.DataSource = listareas;
ddl_area.DataTextField = "Area";
ddl_area.DataValueField = "Id_area";
ddl_area.DataBind();

//ddl PROVINCIAS
List<PROVINCIAS> listprovincias = new List<PROVINCIAS>();
GAC_Provincias_10_Logica datos3 = new GAC_Provincias_10_Logica();
listprovincias = datos3.getProvincias();
ddl_Provincia.DataSource = listprovincias;
ddl_Provincia.DataTextField = "Provincia";
ddl_Provincia.DataValueField = "Id_provincia";
ddl_Provincia.DataBind();

//ddl Cantones
List<CANTONES> listcantones = new List<CANTONES>();
GAC_Cantones_10_Logica datos4 = new GAC_Cantones_10_Logica();
listcantones = datos4.getCantones();
ddl_Canton.DataSource = listcantones;
ddl_Canton.DataTextField = "Canton";
ddl_Canton.DataValueField = "Id_canton";
ddl_Canton.DataBind();

```

114. Ejecutamos el proyecto, y deberá salir algo idéntico a la siguiente imagen

Mi aplicación ASP.NET

localhost:44302/newHospitales

Aplicaciones Gmail YouTube Maps

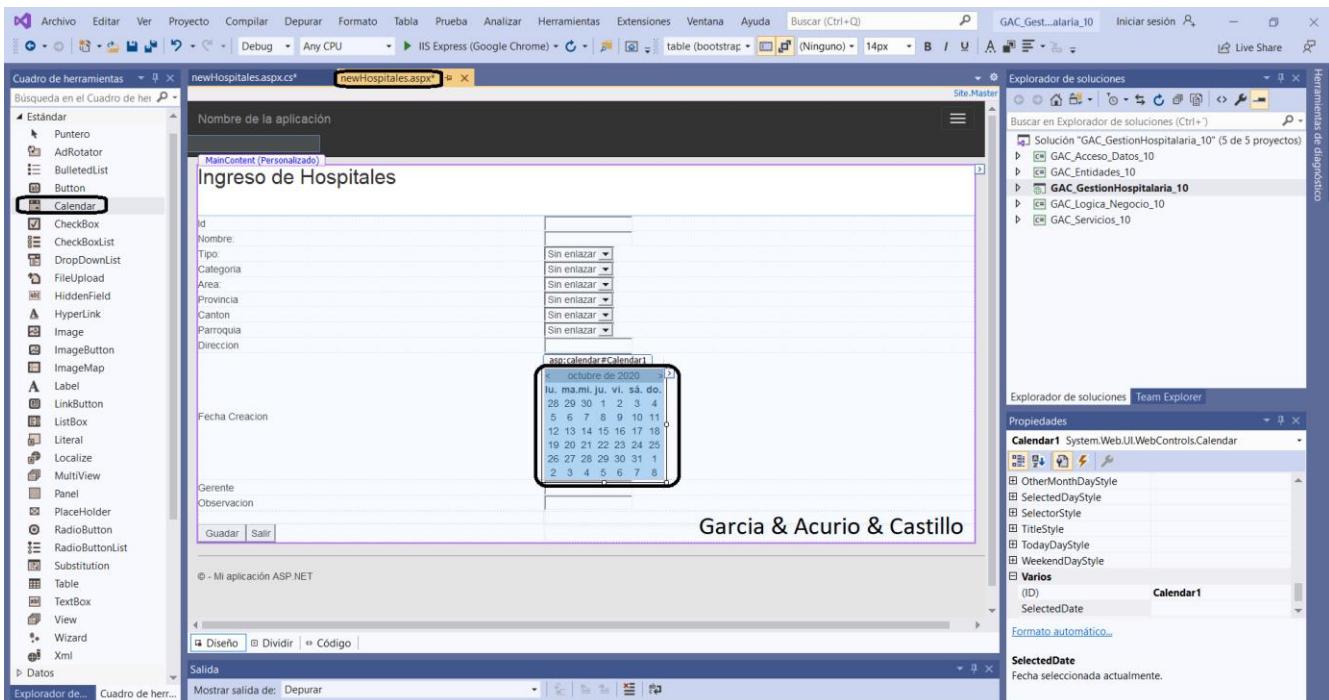
Nombre de la aplicación Inicio Acerca de Contacto

Ingreso de Hospitales

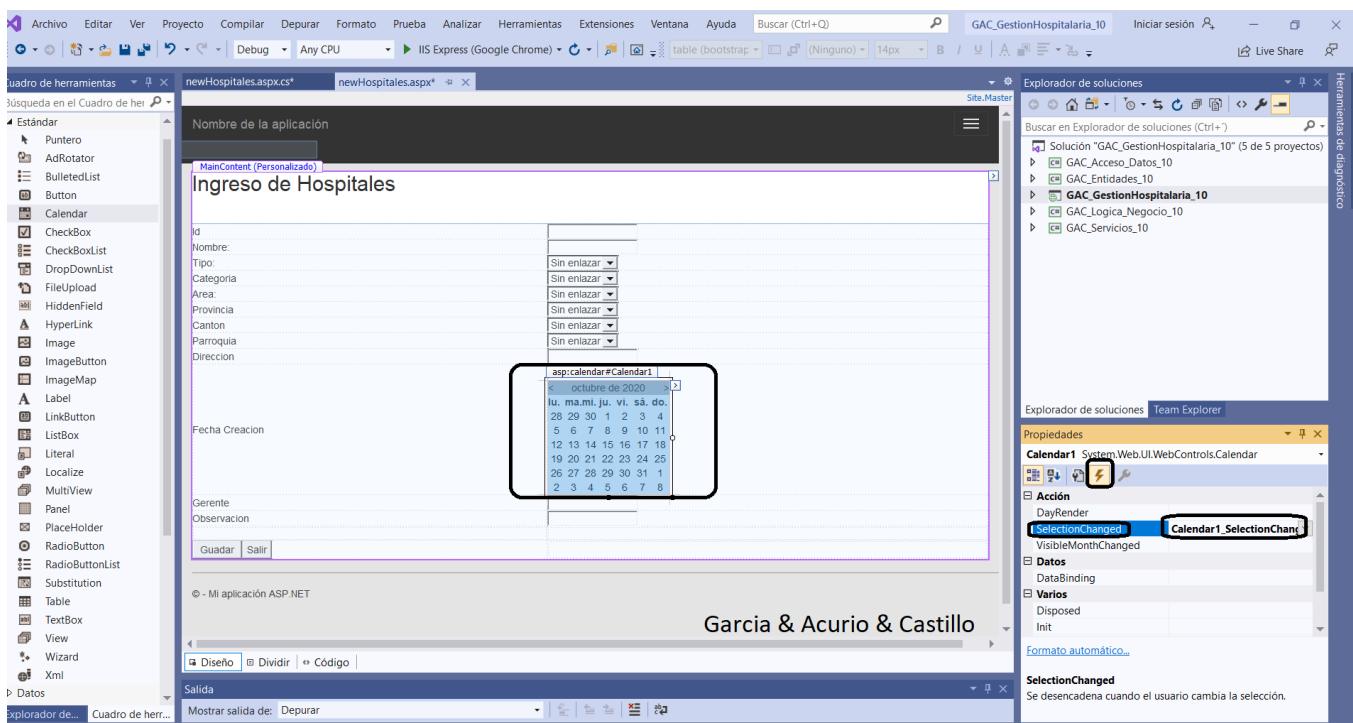
Id	
Nombre:	
Tipo:	Hospital
Categoría	A
Area:	Urbano Central
Provincia	Esmeraldas
Canton	Esmeraldas
Parroquia	Luis Tello
Direccion	
Fecha Creacion	
Gerente	
Observacion	
<input type="button" value="Guardar"/>	<input type="button" value="Salir"/>

Garcia & Acurio & Castillo

115. Agregamos un control tipo calendario y lo pegamos a lado de “txt_Fecha_Creacion”



116. Luego de eso seleccionamos el calendario, nos dirigimos al evento “SelectionChanged” y le damos doble clic.



117. Luego se nos abre una venta donde vamos a escribir dentro del evento la siguiente codificación

```

newHospitales.aspx.cs
newHospitales.aspx*
GAC_GestionHospitalaria_10
Calendar1_SelectionChanged(object sender, EventArgs e)
    {
        txt_Fecha_Creacion.Text = Calendar1.SelectedDate.Date.ToShortDateString();
    }

```

118. Luego ejecutamos y nos debe salir una ventana con la fecha en este modelo como esta en la imagen

Nombre de la aplicación Inicio Acerca de Contacto

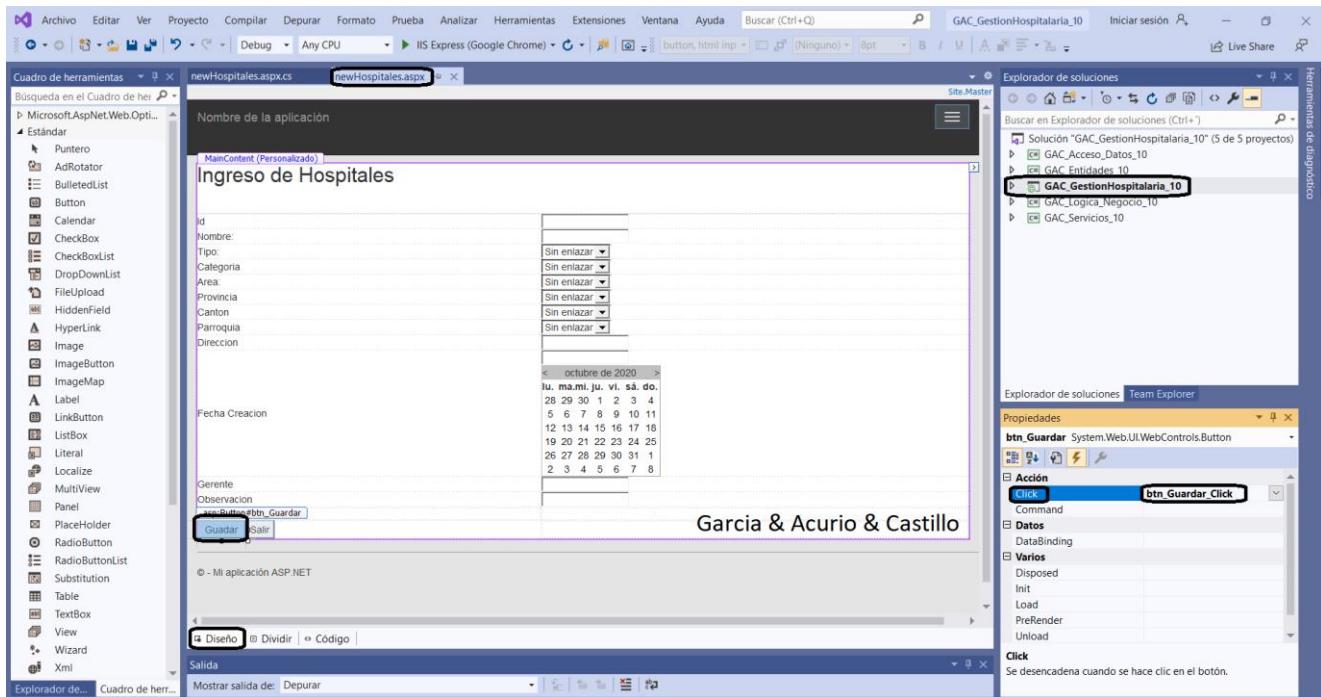
Ingreso de Hospitales

Id	<input type="text"/>
Nombre:	<input type="text"/>
Tipo:	Hospital
Categoría	A
Área:	Urbano Central
Provincia	Esmerealdas
Canton	Esmerealdas
Parroquia	Luis Tello
Dirección	<input type="text"/>
Fecha Creación	08/10/2020
Gerente	<input type="text"/>
Observación	<input type="text"/>
<input type="button" value="Guardar"/> <input type="button" value="Salir"/>	

Garcia & Acurio & Castillo

3.14. CODIFICAR EL BOTÓN GUARDAR

119. Nos dirigimos al archivo “newHospitales.aspx”, estando en el archivo nos dirigimos a **diseño**, en el botón **Guardar** y hacemos doble Click sobre el evento



120. Luego dentro del evento del botón guardar escribimos el siguiente código para poder guardar todos los parámetros

```

protected void btn_Guardar_Click(object sender, EventArgs e)
{
    HOSPITALES nuevoHospital = new HOSPITALES();
    nuevoHospital.Nombre = txt_Nombre.Text;
    nuevoHospital.Tipos = Convert.ToInt32(ddl_Tipo.SelectedValue);
    nuevoHospital.Categoría = Convert.ToInt32(ddl_Categorías.SelectedValue);
    nuevoHospital.Area = Convert.ToInt32(ddl_area.SelectedValue);
    nuevoHospital.Provincia = Convert.ToInt32(ddl_Provincia.SelectedValue);
    nuevoHospital.Canton = Convert.ToInt32(ddl_Canton.SelectedValue);
    nuevoHospital.Parroquia = Convert.ToInt32(ddl_Parroquia.SelectedValue);
    nuevoHospital.Dirección = txt_Dirección.Text;
    nuevoHospital.Fecha_Creacion = Convert.ToDateTime(txt_Fecha_Creacion.Text);
    nuevoHospital.Gerente = txt_Gerente.Text;
    nuevoHospital.Observacion = txt_Observacion.Text;

    GAC_Hospitales_10_Lógica datos = new GAC_Hospitales_10_Lógica();
    datos.NewHospital(nuevoHospital);
}

```

121. Luego ejecutamos y probamos el botón **Guardar**

Nombre de la aplicación Inicio Acerca de Contacto

Ingreso de Hospitales

Id	Simon Bolivar
Nombre:	Subcentro
Tipo:	B
Categoría	Urbano Marginal
Área:	Esmeraldas
Provincia	Esmeraldas
Cantón	Bartolome Ruiz
Parroquia	Calle Bolívar y Juan Montalv
Dirección	15/10/2020
Fecha Creación	< octubre de 2020 > lu. ma.mi. ju. vi. sá. do. 28 29 30 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 1 2 3 4 5 6 7 8
Gerente	Dr. Luis Mendoza
Observación	Esta recién inaugurado
Guardar Salir	

Garcia & Acurio & Castillo

© 2020 - Mi aplicación ASP.NET

122. Abrir la base de datos y verificar que se allá guardado todo correctamente

```

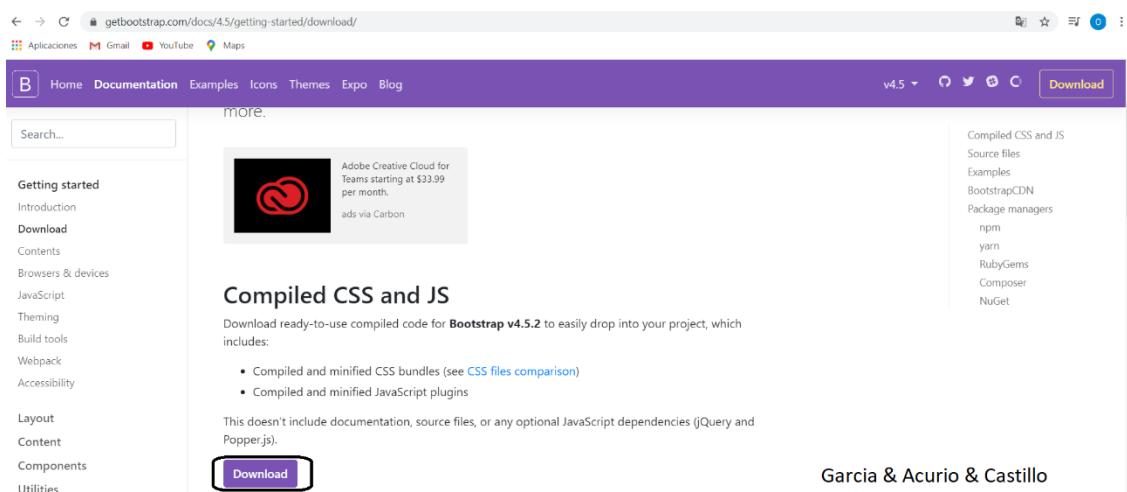
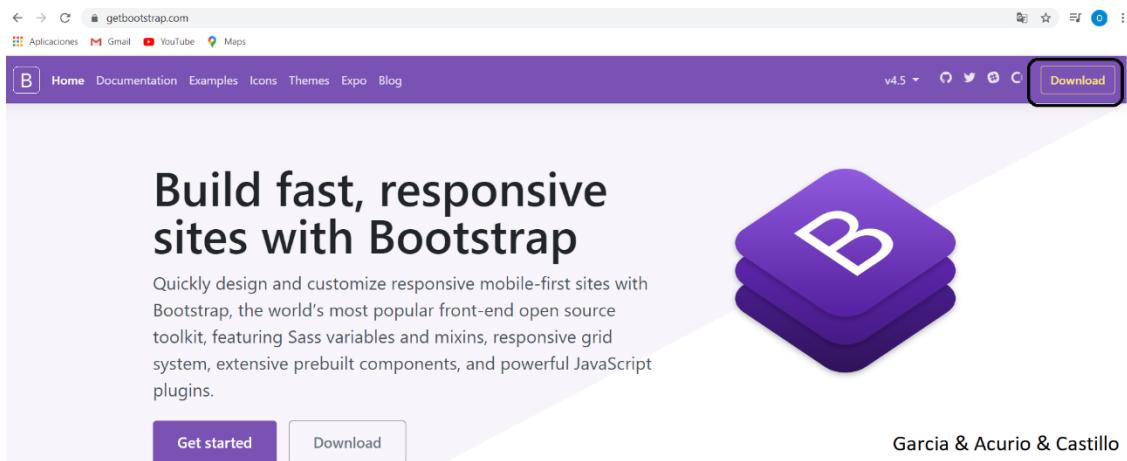
SELECT TOP (1000) [Id_hospital]
      ,[Nombre]
      ,[Tipo]
      ,[Categoría]
      ,[Área]
      ,[Provincia]
      ,[Cantón]
      ,[Parroquia]
      ,[Dirección]
      ,[Fecha_Creación]
      ,[Gerente]
      ,[Observación]
  FROM [GestionHospitalaria].[dbo].[HOSPITALES]
  
```

Id_hospital	Nombre	Tipo	Categoría	Área	Provincia	Cantón	Parroquia	Dirección	Fecha_Creación	Gerente	Observación
1	San Gregorio Loco	3	4	1	1	4	9	Calle Olmedo y Rocafuerte	2018-07-12	Dr. Luis Alba	Esta totalmente Renovado
2	Defina Torres de Concha	1	1	1	9	1	1	Calle Olmedo y Piedrahita	2019-09-06	Dr. Julio Guayasamin	Hospital nuevo
3	Claudia Mejia	5	3	2	20	13	1	Barrio Propicia2	2019-09-12	Saul Gonzalez	ESTA recien
4	Simon Bolívar	4	2	2	1	1	7	Calle Bolívar y Juan Montalvo	2020-10-15	Dr. Luis Mendoza	Esta recien inaugurado

Garcia & Acurio & Castillo

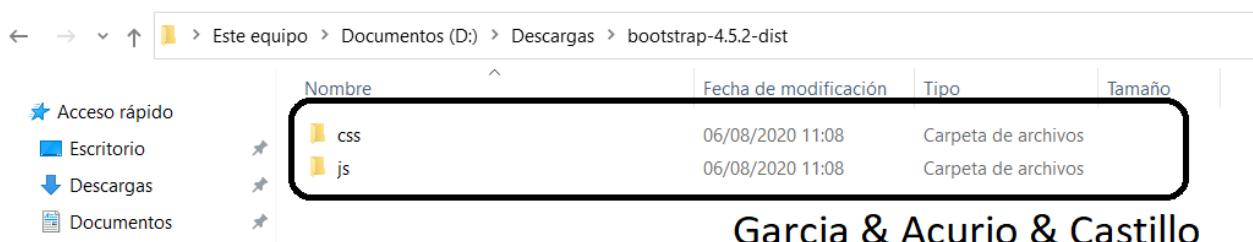
3.15. DISEÑO CON BOOTSTRAP

123. Nos dirigimos al navegador, escribimos en el buscador **Bootstrap** y se nos despliega una venta la cual le damos clic en **Download** y elegimos la opción **Compiled CSS and JS** y le damos en **Download**

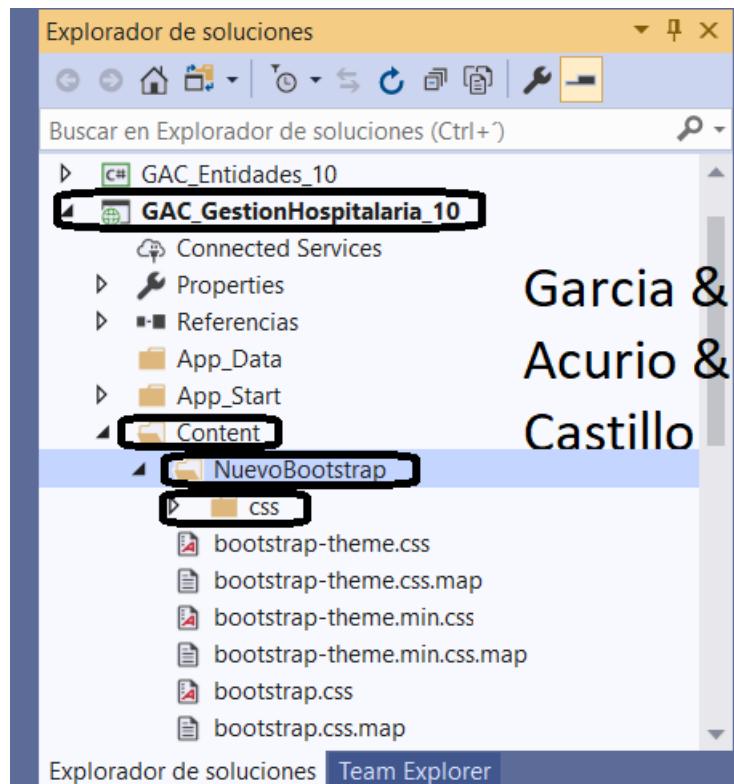


124. Una vez descargado hacemos de descomprimir el documento

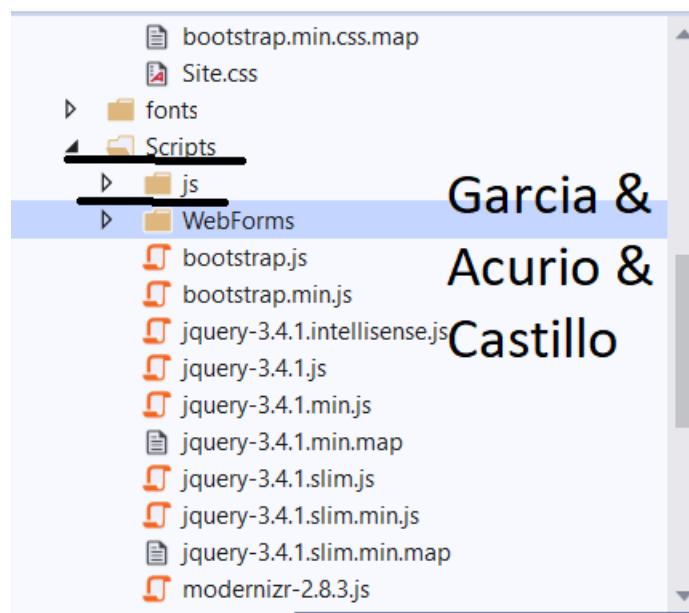
125. Dentro de la carpeta descomprimida hay dos carpetas las cuales hay que añadirlas al proyecto



126. Luego nos dirigimos a la Capa “**GAC_GestionHospitalaria**”, nos vamos a la carpeta **Content** y dentro de eso creamos una nueva carpeta con nombre **NuevoBootstrap** y arrastramos la carpeta **CSS** de la carpeta que descomprimimos, nos debe quedar como en la imagen



127. Luego dentro de la misma Capa “**GAC_GestionHospitalaria**”, nos vamos a la carpeta **Scripts** y arrastramos la carpeta **JS** de la carpeta que descomprimimos, nos debe quedar como en la imagen



128. Una vez echo lo anterior nos dirigimos al **Classroom** y copiamos el código que nos dejaran hay

Jhony Quiñonez
24 sept.

```
<link href="~/Content/Nuevobootstrap4/css/bootstrap.min.css" rel="stylesheet" media="screen"/>
<script src="~/Scripts/jquery-3.3.1.js"></script>
<script src="~/Scripts/js/bootstrap.min.js"></script>
```

Garcia & Acurio & Castillo

129. Una vez ya echo eso nos vamos al archivo “**Siste.Master**” que se encuentra dentro de la Capa “**GAC_GestionHospitalaria_10**”, en la línea 10 damos un enter y pegamos el código que copiamos del classroom debe quedar como en la imagen

Archivo Editar Ver Proyecto Compilar Depurar Prueba Analizar Herramientas Extensiones Ventana Ayuda Buscar (Ctrl+Q) GAC_GestionHospitalaria_10 Iniciar sesión Live Share DOCTYPE: HTML

Cuadro de herramientas Búsqueda en el Cuadro de herramientas Microsoft.AspNet.Web.Optimiz... Estándar Datos Validación Navegación Inicio de sesión Elementos web Extensiones AJAX Datos dinámicos HTML General

No hay controles utilizables en este grupo. Arrastra un elemento a este texto y agréguelo al cuadro de herramientas.

```
<%@ Master Language="C#" AutoEventWireup="true" CodeBehind="Site.master.cs" Inherits="GAC_GestionHospitalaria_10.SiteMaster" %>
<!DOCTYPE html>
<html lang="es">
<head runat="server">
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>%: Page.Title -Gestion Hospitalaria 2020 </title>
    <link href="~/Content/Nuevobootstrap4/css/bootstrap.min.css" rel="stylesheet" media="screen"/>
    <script src="~/Scripts/jquery-3.3.1.js"></script>
    <script src="~/Scripts/js/bootstrap.min.js"></script>
</head>
<body>
    <asp:PlaceHolder runat="server">
        <%: Scripts.Render("~/bundles/modernizr") %>
    </asp:PlaceHolder>
    <webopt:bundlereference runat="server" path "~/Content/css" />
    <link href="~/favicon.ico" rel="shortcut icon" type="image/x-icon" />
</body>
```

Explorador de soluciones Buscar en Explorador de soluciones (Ctrl+) fonts Scripts About.aspx Bundle.config Contact.aspx Default.aspx favicon.ico Global.aspx newHospitales.aspx packages.config Site.Master Site.Mobile.Master ViewSwitcher.ascx Web.config GAC_Logica_Negocio_10

Propiedades <HEAD>

Garcia & Acurio & Castillo

130. Luego modificamos ese código con el nombre nuestra **Carpetas NuevoBootstrap** y la versión de **JQuery**

Archivo Editar Ver Proyecto Compilar Depurar Prueba Analizar Herramientas Extensiones Ventana Ayuda Buscar (Ctrl+Q) GAC_GestionHospitalaria_10 Iniciar sesión Live Share DOCTYPE: HTML

Cuadro de herramientas Búsqueda en el Cuadro de herramientas Microsoft.AspNet.Web.Optimiz... Estándar Datos Validación Navegación Inicio de sesión Elementos web Extensiones AJAX Datos dinámicos HTML General

No hay controles utilizables en este grupo. Arrastra un elemento a este texto y agréguelo al cuadro de herramientas.

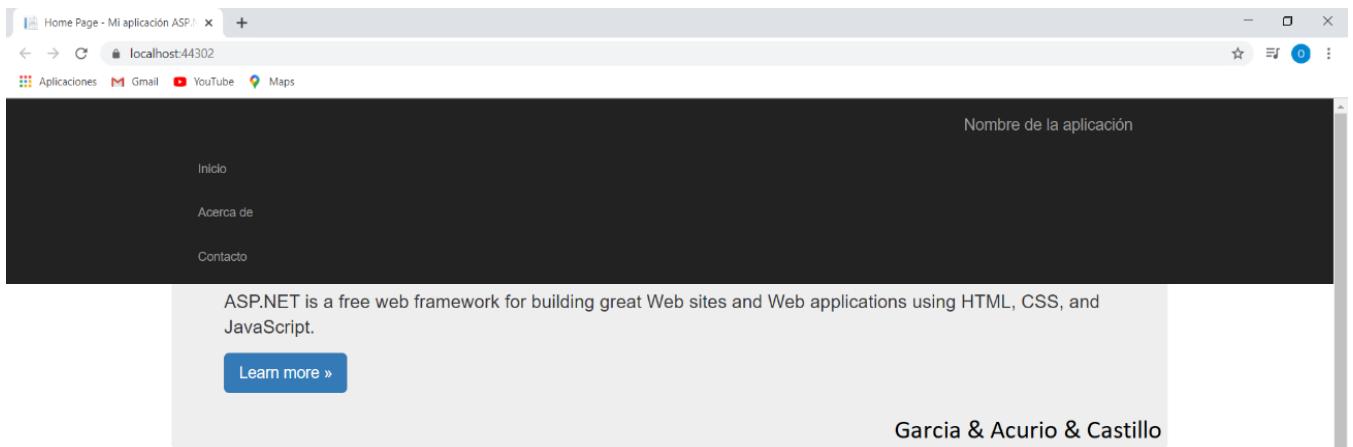
```
<%@ Master Language="C#" AutoEventWireup="true" CodeBehind="Site.master.cs" Inherits="GAC_GestionHospitalaria_10.SiteMaster" %>
<!DOCTYPE html>
<html lang="es">
<head runat="server">
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>%: Page.Title -Gestion Hospitalaria 2020 </title>
    <link href="~/Content/Nuevobootstrap4/css/bootstrap.min.css" rel="stylesheet" media="screen"/>
    <script src="~/Scripts/jquery-3.4.1.js"></script>
    <script src="~/Scripts/js/bootstrap.min.js"></script>
</head>
<body>
    <asp:PlaceHolder runat="server">
        <%: Scripts.Render("~/bundles/modernizr") %>
    </asp:PlaceHolder>
    <webopt:bundlereference runat="server" path "~/Content/css" />
    <link href="~/favicon.ico" rel="shortcut icon" type="image/x-icon" />
</body>
```

Explorador de soluciones Buscar en Explorador de soluciones (Ctrl+) fonts Scripts About.aspx Bundle.config Contact.aspx Default.aspx favicon.ico Global.aspx newHospitales.aspx packages.config Site.Master Site.Mobile.Master ViewSwitcher.ascx Web.config GAC_Logica_Negocio_10

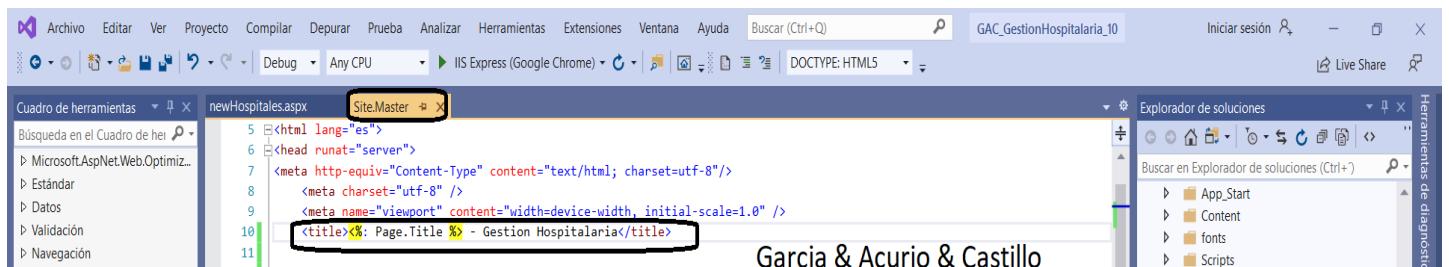
Propiedades GAC_Entidades_10 Propiedades del proyecto

Garcia & Acurio & Castillo

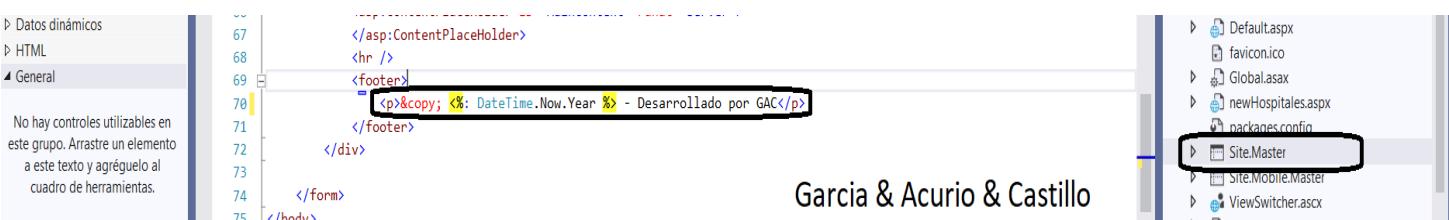
131. Luego ejecutamos y nos deberá quedar, así como se muestra en la imagen



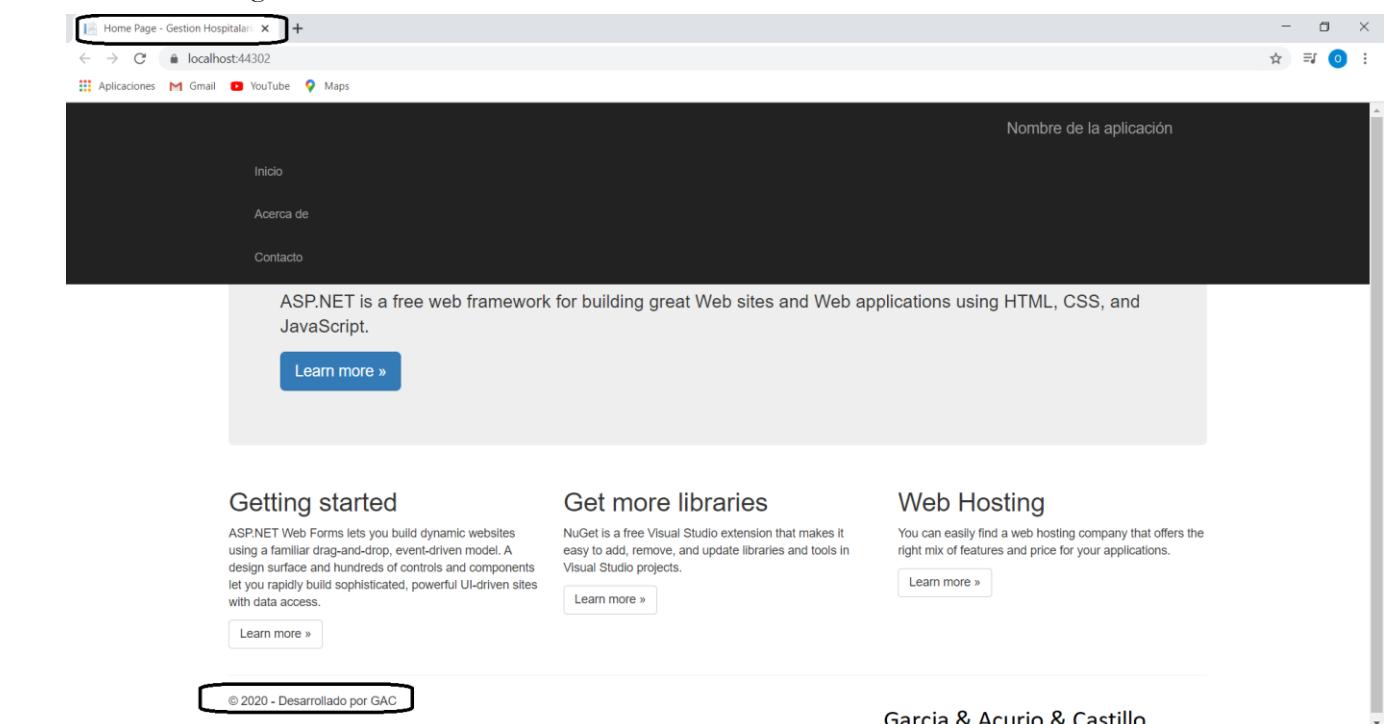
132. Después en el “Siste.Master”, le cambiamos el título de la página



133. Después en el “Siste.Master”, le cambiamos el pie de la página

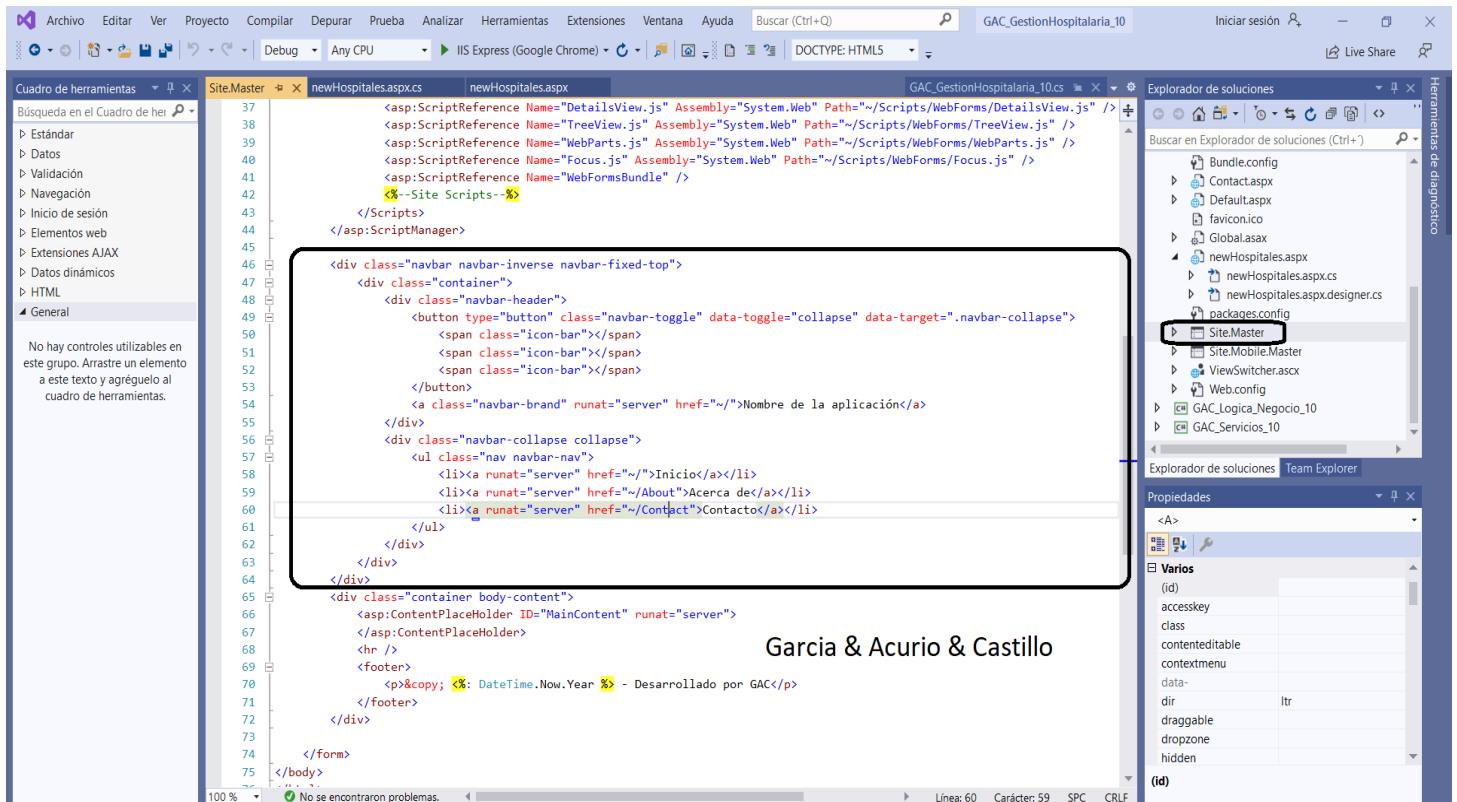


134. Luego ejecutamos para ver los cambios realizados y deberá quedarle como en la imagen



3.16. CAMBIAR EL MODELO DEL MENÚ

135. Dirigirse al “Site.Master” de la capa “GAC_GestionHospitalaria_10” y eliminar desde la línea 46 a la 64



```
<asp:ScriptReference Name="DetailsView.js" Assembly="System.Web" Path "~/Scripts/WebForms/DetailsView.js" />
<asp:ScriptReference Name="TreeView.js" Assembly="System.Web" Path "~/Scripts/WebForms/TreeView.js" />
<asp:ScriptReference Name="WebParts.js" Assembly="System.Web" Path "~/Scripts/WebForms/WebParts.js" />
<asp:ScriptReference Name="Focus.js" Assembly="System.Web" Path "~/Scripts/WebForms/Focus.js" />
<asp:ScriptReference Name="WebFormsBundle" />
<%-- Site Scripts--%>
</Scripts>
</asp:ScriptManager>

<div class="navbar navbar-inverse navbar-fixed-top">
    <div class="container">
        <div class="navbar-header">
            <button type="button" class="navbar-toggle" data-toggle="collapse" data-target=".navbar-collapse">
                <span class="icon-bar"></span>
                <span class="icon-bar"></span>
                <span class="icon-bar"></span>
            </button>
            <a class="navbar-brand" runat="server" href="#">Nombre de la aplicación</a>
        </div>
        <div class="navbar-collapse collapse">
            <ul class="nav navbar-nav">
                <li><a runat="server" href="#">Inicio</a></li>
                <li><a runat="server" href="#">About</a></li>
                <li><a runat="server" href="#">Contact</a></li>
            </ul>
        </div>
    </div>
</div>
<div class="container body-content">
    <asp:ContentPlaceHolder ID="MainContent" runat="server">
    </asp:ContentPlaceHolder>
    <hr />
    <footer>
        <p>&copy; <span id="Year"></span> - Desarrollado por GAC</p>
    </footer>
</div>
</form>
</body>
</html>
```

Garcia & Acurio & Castillo

136. Nos dirigimos al Classroom y copiamos el siguiente código que estará alojado hay



Jhonny Quiñonez

24 sept.

<h2>Gestión Hospitalarias<small> Al Servicio de la Comunidad</small></h2>

```
<div class="page-header">
</div>
<nav class="navbar navbar-inverse navbar-static-top">
    <div class="container">
        <div class="navbar-header">
            <a class="navbar-brand" href="#">Inicio</a>
            <a class="navbar-brand" href="#">Medicos</a>
            <a class="navbar-brand" href="#">Hospitales</a>
            <a class="navbar-brand" href="#">Pacientes</a>
            <a class="navbar-brand" href="#">Contactos</a>
            <a class="navbar-brand" href="#">Nosotros</a>
        </div>
    </div>
</nav>
```

Garcia & Acurio & Castillo

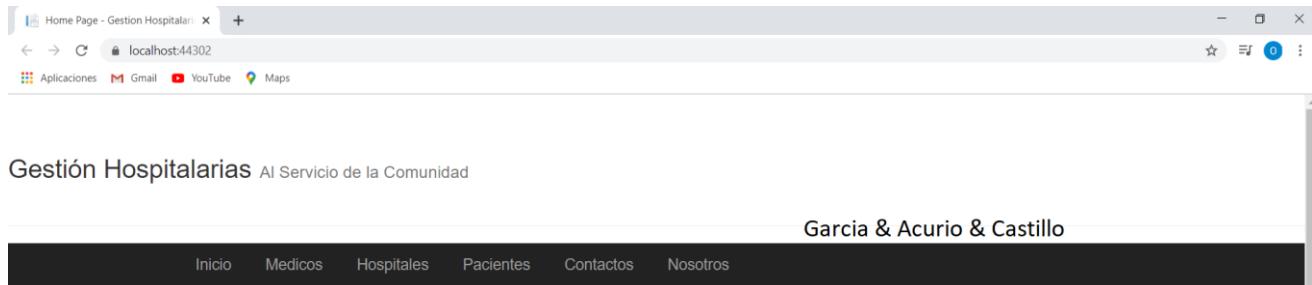
137. Luego nos dirigimos al “Site.Master” de la capa “GAC_GestionHospitalaria_10” y pegamos lo copiado en el Classroom en la línea 46.

```

    <div class="nav-item dropdown">
        <a href="#" class="nav-link dropdown-toggle" data-bs-toggle="dropdown">Nuevo Hospital</a>
        <ul class="dropdown-menu" style="background-color: #f9f9f9; border: 1px solid #ccc; padding: 5px; margin-top: 10px; font-size: 10pt;">
            <li><a href="newHospitales.aspx" style="color: #0072bc; text-decoration: none; font-weight: bold;">Nuevo Hospital</li>
            <li><a href="#" style="color: #0072bc; text-decoration: none; font-weight: bold;">Nuevo Paciente</li>
            <li><a href="#" style="color: #0072bc; text-decoration: none; font-weight: bold;">Nuevo Medico</li>
            <li><a href="#" style="color: #0072bc; text-decoration: none; font-weight: bold;">Nuevo Hospital</li>
        </ul>
    </div>

```

138. Ejecutamos y el menú deberá quedar como en la imagen.



3.17. CAMBIAR EL MODELO DE NEWHOSPITALES

139. Nos dirigimos al Classroom y copiamos el código alojado hay

Jhony Quiñonez
24 sept.

```

<div class="p-1">
    <div class="input-group">
        <span class="input-group-addon" id="span1">Código:</span>
        <asp:TextBox ID="txt_HospitalID" runat="server" placeholder="Ingrese el código" CssClass="form-control">
    </asp:TextBox>
</div>
</div>

```

Garcia & Acurio & Castillo

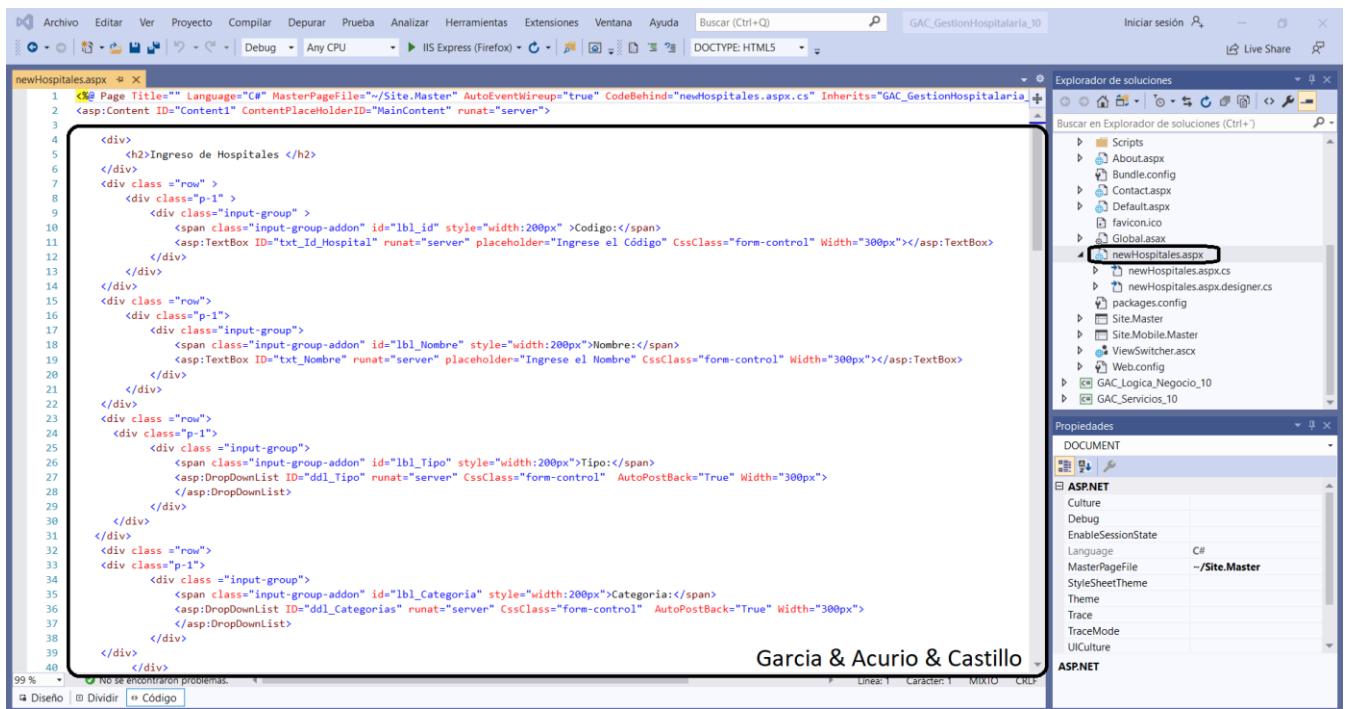
140. Nos dirigimos a “newHospitales.aspx” y comenzamos en el código pegamos lo copiado del Classroom y comenzamos a modificar



```
<%@ Page Title="" Language="C#" MasterPageFile("~/Site.Master" AutoEventWireup="true" CodeBehind="newHospitales.aspx.cs" Inherits="GAC_GestionHospitalaria_10.newHospitales" %>
<asp:Content ID="Content1" ContentPlaceHolderID="MainContent" runat="server">
    <div>
        <h2>Ingreso de Hospitales </h2>
    </div>
    <div class="p-1">
        <div class="input-group">
            <span class="input-group-addon" id="span1">Código:</span>
            <asp:TextBox ID="txt_HospitalID" runat="server" placeholder="Ingrese el código" CssClass="form-control"></asp:TextBox>
        </div>
    </div>
    <div>
        <h2>Ingreso de Hospitales </h2>
        <p>&ampnbsp</p>
        <table style="width: 100%">
            <tr>
                <td id="lbl_id"> Id</td>
                <td>
                    <asp:TextBox ID="txt_Id_Hospital" runat="server" ></asp:TextBox>
                </td>
            </tr>
            <tr>
                <td id="lbl_Nombre"> Nombre:</td>
                <td>
```

Garcia & Acurio & Castillo

141. Luego de modificar nos deberá quedar, así como en la imagen



```
<%@ Page Title="" Language="C#" MasterPageFile "~/Site.Master" AutoEventWireup="true" CodeBehind="newHospitales.aspx.cs" Inherits="GAC_GestionHospitalaria_10.newHospitales" %>
<asp:Content ID="Content1" ContentPlaceHolderID="MainContent" runat="server">
    <div>
        <h2>Ingreso de Hospitales </h2>
    </div>
    <div class="row" >
        <div class="p-1" >
            <div class="input-group" >
                <span class="input-group-addon" id="lbl_id" style="width:200px" >Código:</span>
                <asp:TextBox ID="txt_Id_Hospital" runat="server" placeholder="Ingrese el Código" CssClass="form-control" Width="300px" ></asp:TextBox>
            </div>
        </div>
        <div class="row" >
            <div class="p-1" >
                <div class="input-group" >
                    <span class="input-group-addon" id="lbl_Nombre" style="width:200px" >Nombre:</span>
                    <asp:TextBox ID="txt_Nombre" runat="server" placeholder="Ingrese el Nombre" CssClass="form-control" Width="300px" ></asp:TextBox>
                </div>
            </div>
            <div class="row" >
                <div class="p-1" >
                    <div class="input-group" >
                        <span class="input-group-addon" id="lbl_Tipo" style="width:200px" >Tipo:</span>
                        <asp:DropDownList ID="ddl_Tipo" runat="server" CssClass="form-control" AutoPostBack="True" Width="300px" ></asp:DropDownList>
                    </div>
                </div>
                <div class="row" >
                    <div class="p-1" >
                        <div class="input-group" >
                            <span class="input-group-addon" id="lbl_Categoría" style="width:200px" >Categoria:</span>
                            <asp:DropDownList ID="ddl_Categorías" runat="server" CssClass="form-control" AutoPostBack="True" Width="300px" ></asp:DropDownList>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>
</asp:Content>
```

Garcia & Acurio & Castillo

142. Despu s copiamos otro c digo del Classroom para modificar el formato de la Fecha

Jhonny Qui onez
24 sept.

```
<div class="input-group" style="left: 0px; top: 0px">
    <span class="input-group-addon" id="span10">F. Creacion:</span>
    <asp:TextBox ID="txt_Fcreacion" runat="server" placeholder="Ingrese F creacion" CssClass="form-control">
</asp:TextBox>
    <asp:Calendar ID="cal_FechaCreacion" runat="server" SelectedDate="07/22/2018 14:50:51"
OnSelectionChanged="cal_FechaCreacion_SelectionChanged"></asp:Calendar>
</div>
```

Garcia & Acurio & Castillo

143. Nos dirigimos a “newHospitales.aspx” y pegamos el c digo del Classroom y modificamos

```
14
15
16
17 <div class="p-1">
18     <div class="input-group">
19         <span class="input-group-addon" id="span3">Tipo hospital:</span>
20         <asp:DropDownList ID="ddl_TipoHospital" runat="server" CssClass="form-control" OnSelectedIndexChanged="ddl_TipoHospital_SelectedIndexChanged">
21             </asp:DropDownList>
22     </div>
23 </div>

24
25     <td id="lbl_Fecha_Creacion" style="height: 20px" > Fecha Creacion</td>
26     <td style="height: 20px" >
27         <asp:TextBox ID="txt_Fecha_Creacion" runat="server"></asp:TextBox>
28         <asp:Calendar ID="Calendar1" runat="server" OnSelectionChanged="Calendar1_SelectionChanged"></asp:Calendar>
29     </td>
30
31
```

Garcia & Acurio & Castillo

144. Nos deber  quedo como en la imagen

```
66 <div class = "row">
67     <div class="p-1">
68         <div class="input-group" style="left: 0px; top: 0px">
69             <span class="input-group-addon" id="lbl_Fecha_Creacion" style="width:200px">Fecha Creacion:</span>
70             <asp:TextBox ID="txt_Fecha_Creacion" runat="server" placeholder="Ingrese Fecha Creacion" CssClass="form-control" ></asp:TextBox>
71             <asp:Calendar ID="Calendar1" runat="server" SelectedDate="07/22/2018 14:50:51" OnSelectionChanged="Calendar1_SelectionChanged"></asp:Calendar>
72         </div>
73     </div>
74     </div>
75     <div class = "row">
76         <div class="p-1">
77             <div class="input-group">
78                 <span class="input-group-addon" id="lbl_Gerente" style="width:200px">Gerente:</span>
79                 <asp:TextBox ID="txt_Gerente" runat="server" placeholder="Ingrese el Gerente" CssClass="form-control" Width="300px"></asp:TextBox>
80             </div>
81         </div>
82         <div class = "row">
83             <div class="p-1">
84                 <div class="input-group">
85                     <span class="input-group-addon" id="lbl_Observacion" style="width:200px">Observacion:</span>
86                     <asp:TextBox ID="txt_Observacion" runat="server" placeholder = "Ingrese la Observacion" CssClass="form-control" Width="300px"></asp:TextBox>
87                 </div>
88             </div>
89             <div class = "row">
90                 <div class="p-1">
91                     <div class="btn-group" role = "group" aria-level = "">
92                         <asp:Button ID="Button1" runat="server" Text="Guardar" OnClick="btn_Guardar_Click" CssClass="btn btn-info"/>
93                         <asp:Button ID="Button2" runat="server" Text="Salir" CssClass="btn btn-info" OnClick="Button2_Click"/>
94                     </div>
95                 </div>
96             </div>
97         </div>
98     </div>
99 
```

Garcia & Acurio & Castillo

145. Luego ejecutamos para ver como nos quedara

Ingreso de Hospitales

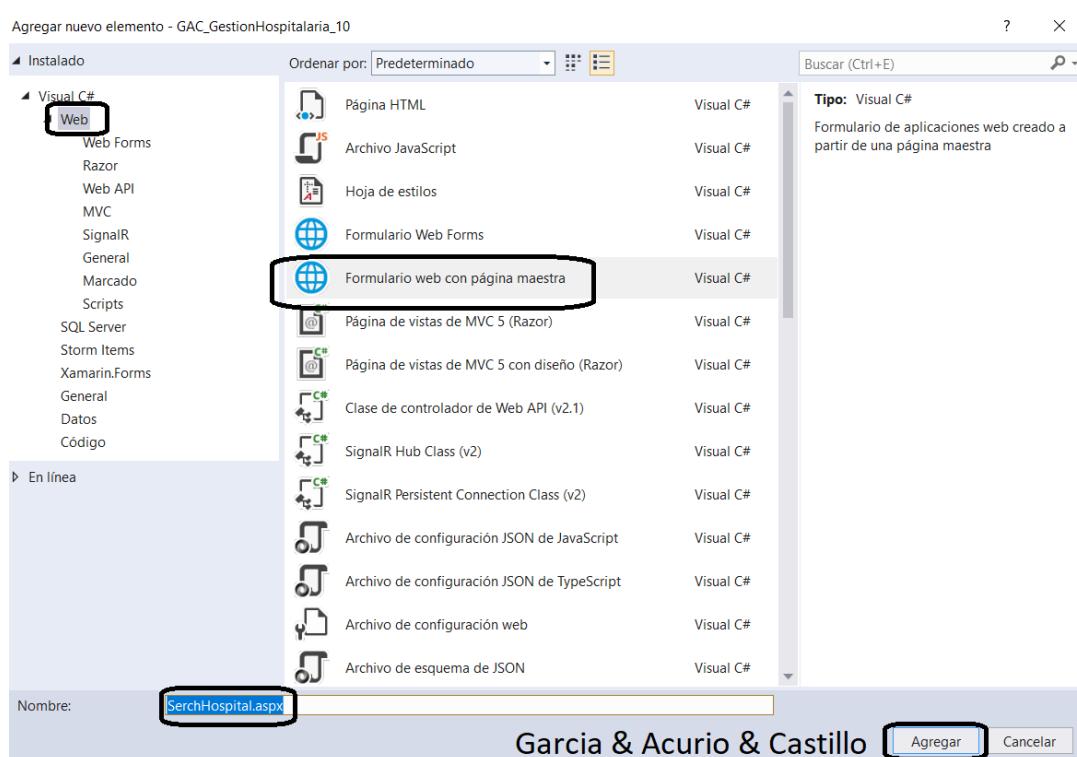
Código:	Ingresel Código
Nombre:	Ingresel Nombre
Tipo:	Hospital
Categoría:	A
Área:	Urbano Central
Provincia:	Esmeraldas
Cantón:	Esmeraldas
Parroquia:	Luis Tello
Dirección:	Ingresel Dirección
Fecha Creación:	Ingresel Fecha Creacion octubre de 2020 lun. mar. mié. jue. vie. sáb. dom. 28 29 30 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 1 2 3 4 5 6 7 8
Gerente:	Ingresel Gerente
Observación:	Ingresel Observacion

Guardar Salir

Garcia & Acurio & Castillo

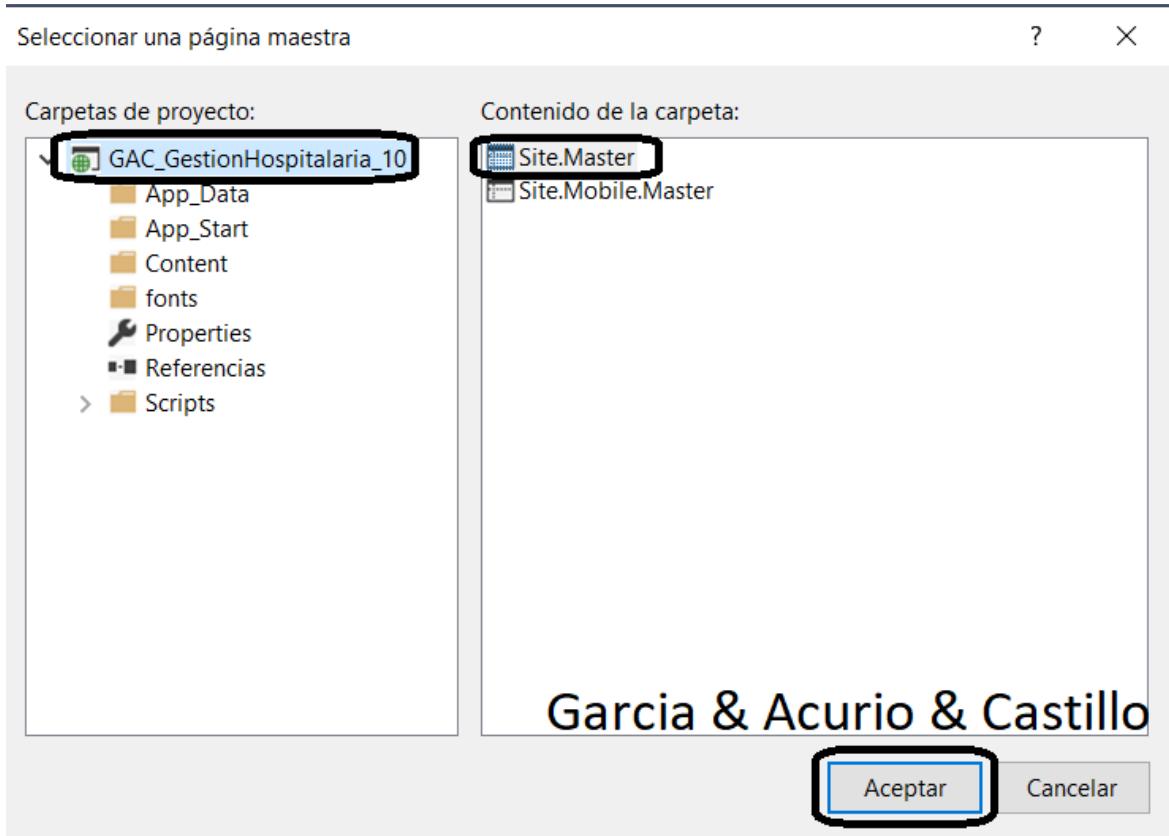
3.18. CREACIÓN DEL DISEÑO DE LA PAGINA BUSCAR HOSPITAL

146. Hacemos clic derecho sobre la capa de presentación y **agregar- nuevo elemento**, elegimos formulario de tipo web con la opción de **página maestra**, colocamos el nombre de **SerchHospital** y agregamos

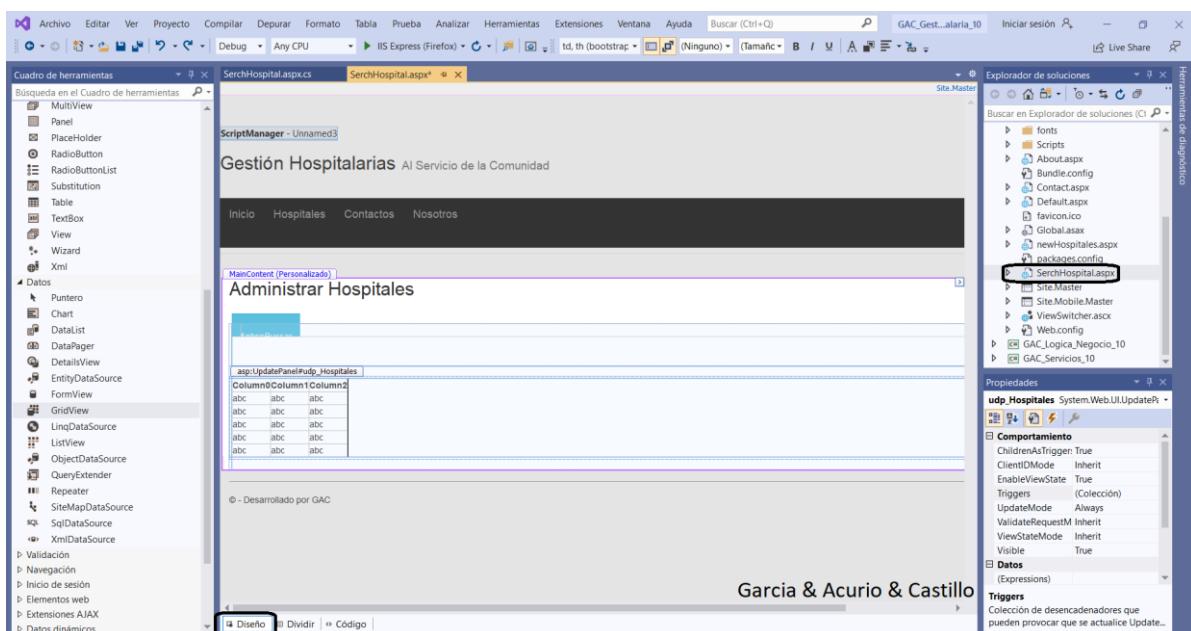


Garcia & Acurio & Castillo

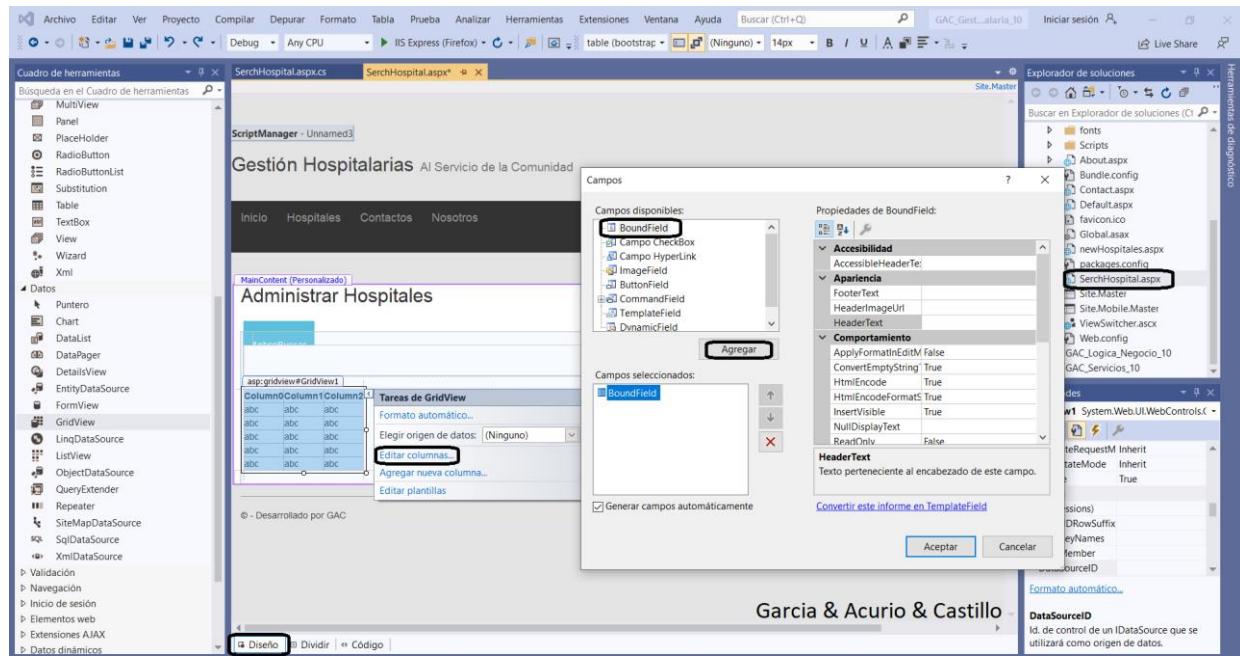
147. Se nos mostrara una ventana, elegimos la opción **Site.Master** y luego aceptamos



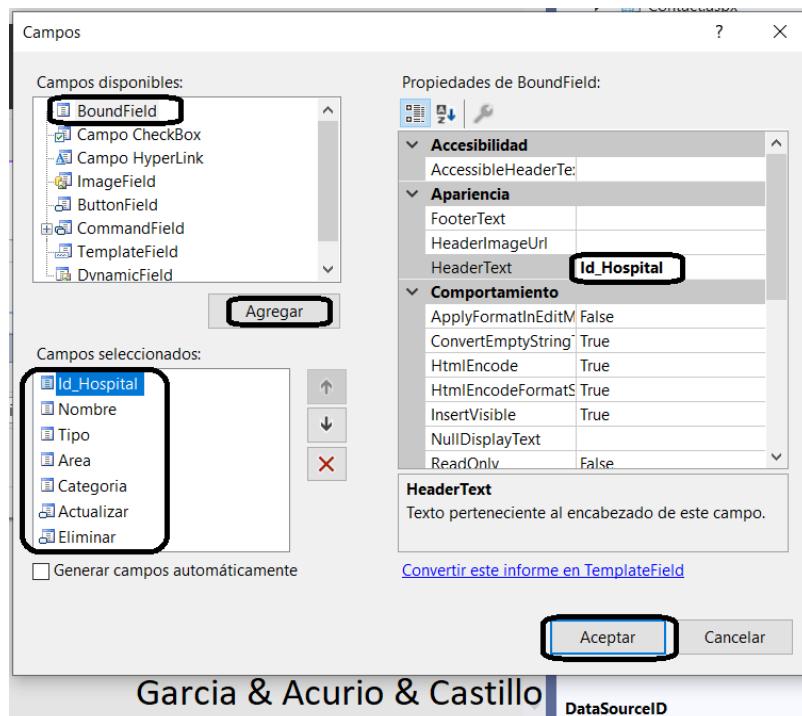
148. Luego de eso dentro del formulario creado en el **diseño** agregamos **nombre** agregamos una **tabla** con **2 columnas y 3 filas**, agregamos **LinkButton** con nombre **lb_Buscar**, agregamos **TextBox** con nombre **txt_Criterio**, agregamos un **UpdatePanel** con nombre **udp_Hospitales**, agregamos un **GridView** con nombre **gv_Hospitales**.



149. Luego en el **GridView** con nombre **gv_Hospitales**, damos en **editar columnas** luego en **BoundField** y agregamos, luego editamos



150. Después modificamos los nombres y agregamos más campos como se muestra en la imagen y le damos en aceptar



151. Una vez aceptado nos debe quedar, así como en la imagen

The screenshot shows a web page titled "Garcia & Acurio & Castillo". At the top left, there is a search bar labeled "Buscar" with a magnifying glass icon. Below the search bar is a table with the following columns: Id_Hospital, Nombre, Tipo, Area, Categoria, Actualizar, and Eliminar. The table has five rows, each containing a "DataBound" placeholder for the respective column.

Id_Hospital	Nombre	Tipo	Area	Categoría	Actualizar	Eliminar
DataBound	DataBound	DataBound	DataBound	DataBound	Actualizar	Eliminar
DataBound	DataBound	DataBound	DataBound	DataBound	Actualizar	Eliminar
DataBound	DataBound	DataBound	DataBound	DataBound	Actualizar	Eliminar
DataBound	DataBound	DataBound	DataBound	DataBound	Actualizar	Eliminar

152. Luego nos vamos al **Classroom** y copiamos el código alojado hay



Jhony Quiñonez
30 sept.

```
<div class="input-group">
    <span class="input-group-btn">
        <asp:LinkButton ID="lb_Buscar" runat="server" CssClass="btn btn-info" Text="Buscar"
        OnClick="lb_Buscar_Click"><span class="glyphicon glyphicon-search"></span>&ampnbspBuscar</asp:LinkButton>
    </span>
    <asp:TextBox ID="txt_Criterio" runat="server" CssClass="form-control"></asp:TextBox>
</div>
```

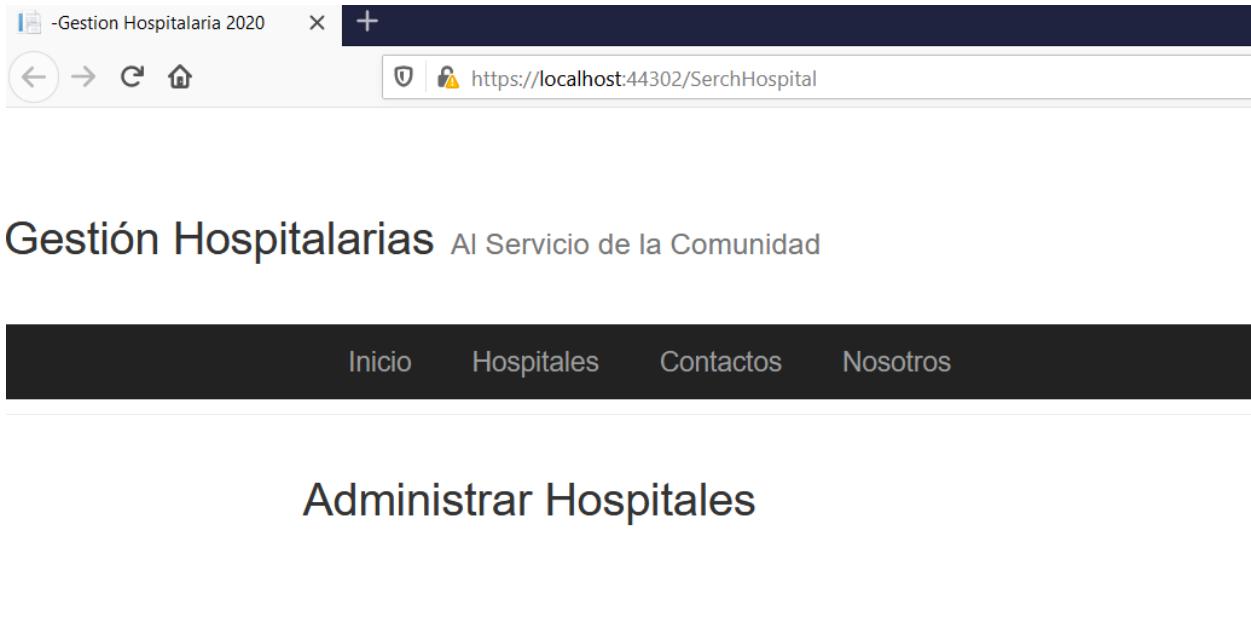
Garcia & Acurio & Castillo

153. Luego nos vamos al **SerchHospital.aspx** y en el código en la línea 14 pegamos lo copiado como se muestra en la imagen

The screenshot shows the Visual Studio IDE with the code editor open for "SerchHospital.aspx.cs". The code is as follows:

```
1 <%@ Page Title="" Language="C#" MasterPageFile("~/Site.Master" AutoEventWireup="true" CodeBehind="SerchHospital.aspx.cs" Inherits="GAC_GestionHospitalaria_10.SerchHospital" %>
2 <%>
3
4     <div>
5         <h2> Administrar Hospitales </h2>
6         <p>&ampnbsp</p>
7     </div>
8
9     <table style="width:100%; height: 235px;">
10        <tr>
11            <td style="width: 60px;">&ampnbsp</td>
12            <td>
13
14                <div class="input-group">
15                    <span class="input-group-btn">
16                        <asp:LinkButton ID="lb_Buscar" runat="server" CssClass="btn btn-info" Text="Buscar"
17                        OnClick="lb_Buscar_Click"><span class="glyphicon glyphicon-search"></span>&ampnbspBuscar</asp:LinkButton>
18                    </span>
19                    <asp:TextBox ID="txt_Criterio" runat="server" CssClass="form-control"></asp:TextBox>
20                </div>
21            </td>
22        </tr>
23        <tr>
24            <td style="width: 60px;">&ampnbsp</td>
25            <td>
26
27                <asp:UpdatePanel ID="upd_Hospitales" runat="server">
28                    <ContentTemplate>
29                        <asp:GridView ID="gv_Hospitales" runat="server" AutoGenerateColumns="False" OnRowCommand="gv_H...
30                            <Columns>
31                                <asp:BoundField DataField="Id_hospital" HeaderText="Id_Hospital" />
32                                <asp:BoundField DataField="Nombre" HeaderText="Nombre" />
33                                <asp:BoundField DataField="Tipo" HeaderText="Tipo" />
34                                <asp:BoundField DataField="Area" HeaderText="Area" />
35                                <asp:BoundField DataField="Categoria" HeaderText="Categoria" />
36                                <asp:ButtonField CommandName="Actualizar" HeaderText="Actualizar" Text="Actualizar" />
37                                <asp:ButtonField CommandName="Eliminar" HeaderText="Eliminar" Text="Eliminar" />
38                            </Columns>
39                        </asp:GridView>
40                    </ContentTemplate>
41                </asp:UpdatePanel>
42            </td>
43        </tr>
44    </table>
45
46    <div style="text-align: right; margin-top: 10px;">
47        <a href="#" class="btn btn-primary" role="button">Nuevo
48        <a href="#" class="btn btn-primary" role="button">Actualizar
49        <a href="#" class="btn btn-primary" role="button">Eliminar
50    </div>
51
52</div>
```

154. Ejecutamos para ver como se ve el diseño



Garcia & Acurio & Castillo

3.19. CODIFICACIÓN DEL BOTÓN BUSCAR

155. Nos vamos al formulario **SerchHospital.aspx.cs**, agregamos las librerías y programamos el botón **lb_Buscar** con el evento **Click**

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5  using System.Web.UI;
6  using System.Web.UI.WebControls;
7  using GAC_Entidades_10;
8  using GAC_Logica_Negocio_10;
9
10 namespace GAC_GestionHospitalaria_10
11 {
12     public partial class SerchHospital : System.Web.UI.Page
13     {
14         protected void Page_Load(object sender, EventArgs e)
15         {
16         }
17
18         protected void lb_Buscar_Click(object sender, EventArgs e)
19         {
20             GAC_Hospitales_10_Logica datos = new GAC_Hospitales_10_Logica();
21             List<HOSPITALES> list hospitales = new List<HOSPITALES>();
22
23             if(txt_Criterio.Text == string.Empty)
24             {
25                 hospitales = datos.getHospitales();
26             }
27             else
28             {
29                 hospitales = datos.getHospitales_id(Convert.ToInt32(txt_Criterio.Text));
30             }
31
32             gv_Hospitales.DataSource = hospitales;
33             gv_Hospitales.DataBind();
34         }
35     }
36 }
37
```

Garcia & Acurio & Castillo

156. Una vez echo el botón buscar ejecutamos y nos debe buscar

Administrar Hospitales

 Buscar

Id_Hospital	Nombre	Tipo	Area	Categoría	Actualizar	Eliminar
1	San Gregorio Loco	2	3		Actualizar	Eliminar
6	Delfina Torres	1	1	1	Actualizar	Eliminar

Garcia & Acurio & Castillo

157. Buscar por **Id**

Administrar Hospitales

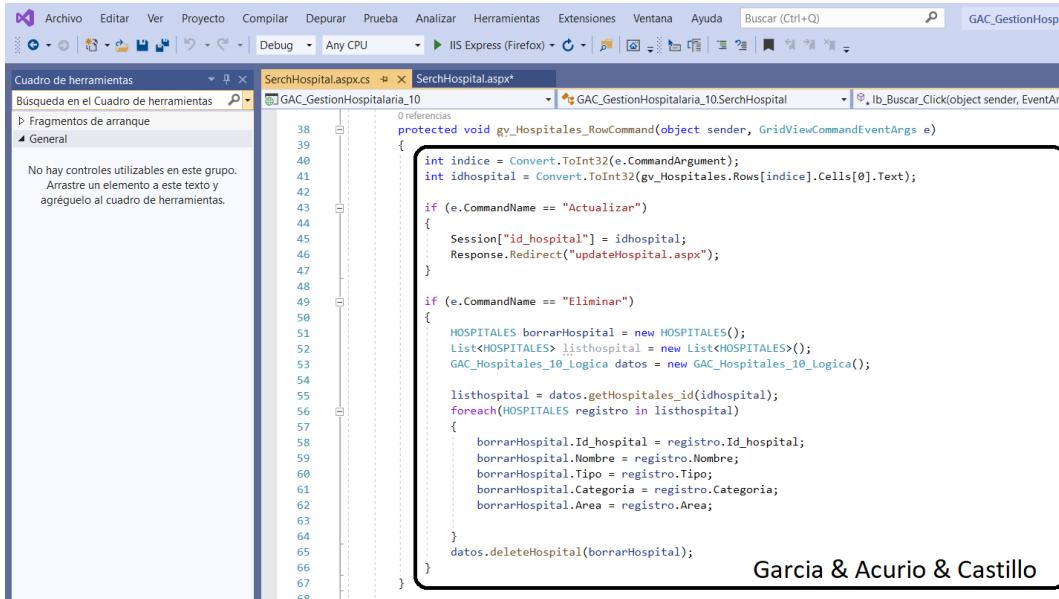
 Buscar

Id_Hospital	Nombre	Tipo	Area	Categoría	Actualizar	Eliminar
6	Delfina Torres	1	1	1	Actualizar	Eliminar

Garcia & Acurio & Castillo

3.20. CODIFICACIÓN DEL BOTÓN ELIMINAR

158. Luego dentro del formulario **SerchHospital.aspx.cs**, programamos el evento **gv_Hospitales_RowCommand**



```
protected void gv_Hospitales_RowCommand(object sender, GridViewCommandEventArgs e)
{
    int indice = Convert.ToInt32(e.CommandArgument);
    int idhospital = Convert.ToInt32(gv_Hospitales.Rows[indice].Cells[0].Text);

    if (e.CommandName == "Actualizar")
    {
        Session["id_hospital"] = idhospital;
        Response.Redirect("updateHospital.aspx");
    }

    if (e.CommandName == "Eliminar")
    {
        HOSPITALES borrarHospital = new HOSPITALES();
        List<HOSPITALES> listospital = new List<HOSPITALES>();
        GAC_Hospitales_10_Logica datos = new GAC_Hospitales_10_Logica();

        listospital = datos.getHospitales_id(idhospital);
        foreach (HOSPITALES registro in listospital)
        {
            borrarHospital.Id_hospital = registro.Id_hospital;
            borrarHospital.Nombre = registro.Nombre;
            borrarHospital.Tipo = registro.Tipo;
            borrarHospital.Categoría = registro.Categoría;
            borrarHospital.Area = registro.Area;
        }
        datos.deleteHospital(borrarHospital);
    }
}
```

Garcia & Acurio & Castillo

159. Luego ejecutamos y eliminamos un archivo buscado y verificamos que elimine

Administrar Hospitales

Buscar

Id_Hospital	Nombre	Tipo	Area	Categoría	Actualizar	Eliminar
1	San Gregorio Loco	2	2	3	Actualizar	Eliminar

Garcia & Acurio & Castillo

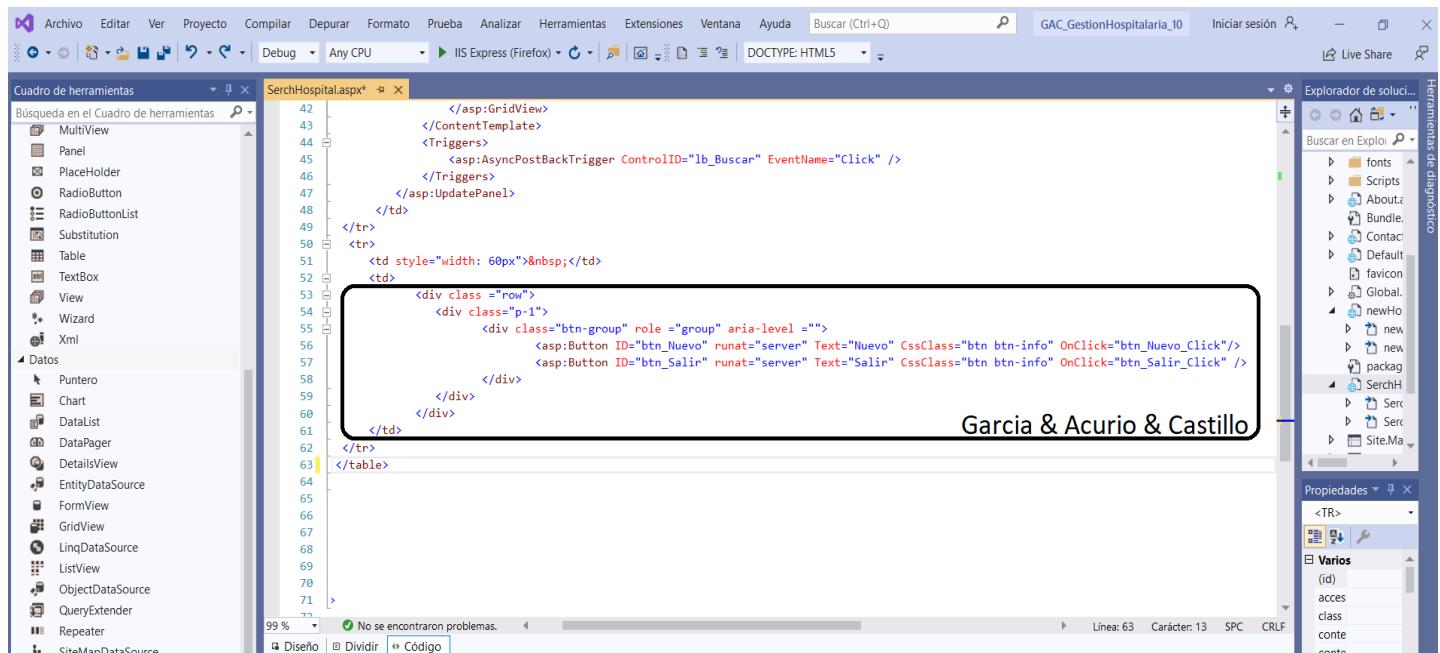
3.21. PROGRAMAR BOTÓN NUEVO Y SALIR DE LA PÁGINA SERCHHOSPITAL

160. Copiamos el código de los botones del formulario newHospitales

```
111 <div class ="row">
112   <div class="p-1">
113     <div class="btn-group" role ="group" aria-level =">
114       <asp:Button ID="Button1" runat="server" Text="Guardar" OnClick="btn_Guardar_Click" CssClass="btn btn-info"/>
115       <asp:Button ID="Button2" runat="server" Text="Salir" CssClass="btn btn-info" OnClick="Button2_Click"/>
116     </div>
117   </div>
118 </div>
119
```

Garcia & Acurio & Castillo

161. Luego una ves copiado modificamos los **Id** de los botones y ejecutamos el evento **onClick** en los botones **Nuevo** y **Salir**



162. Luego ejecutamos y debería quedar como en la imagen

Administrar Hospitales

Buscar

Nuevo

Salir

Garcia & Acurio & Castillo

163. Luego nos vamos al formulario **SerchHospital.aspx.cs**, y en los eventos Programamos en el Evento **onClick** como muestra en la imagen

```
68
69     protected void btn_Nuevo_Click(object sender, EventArgs e)
70     {
71         Response.Redirect("newHospitales.aspx");
72     }
73
74     protected void btn_Salir_Click(object sender, EventArgs e)
75     {
76         Response.Redirect("Default.aspx");
77     }
78
79 }
```

Garcia & Acurio & Castillo

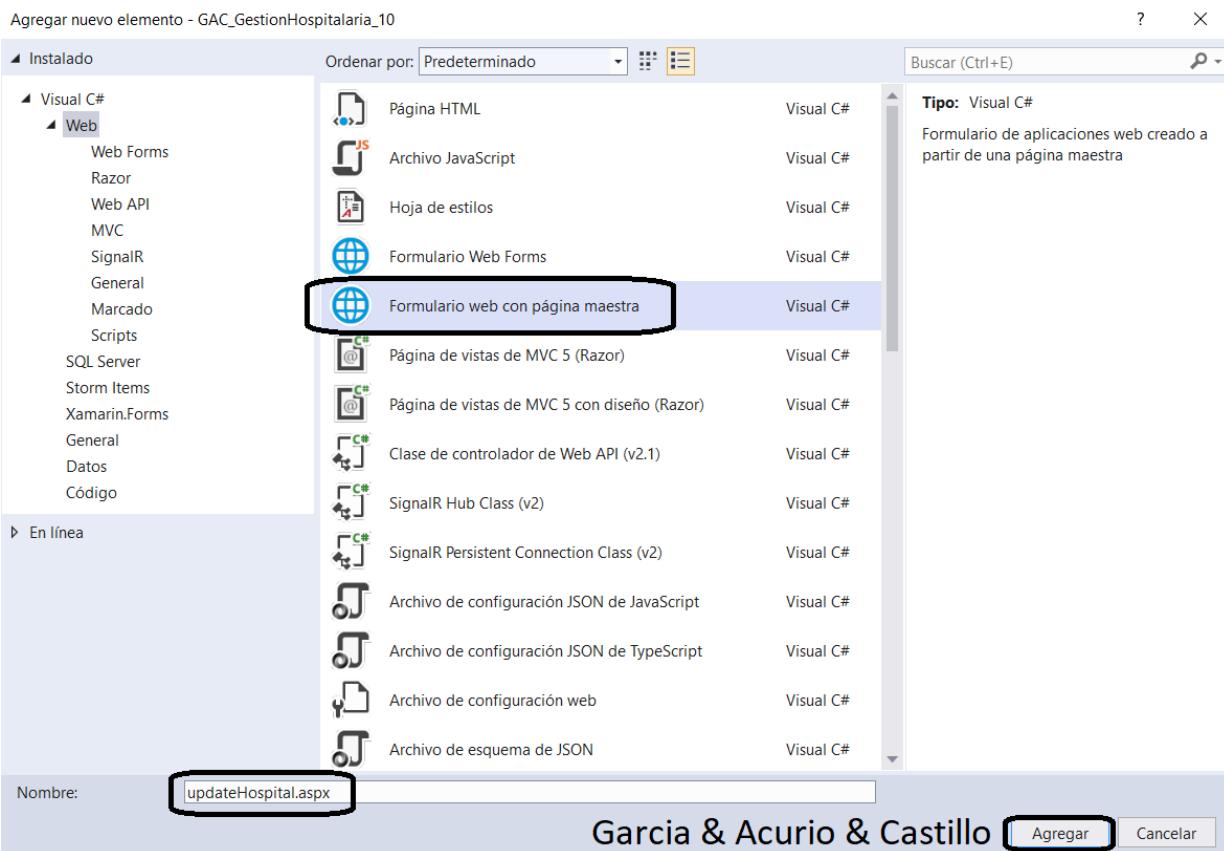
Explorador de soluciones

- fonts
- Scripts
- About.aspx
- Bundle.config
- Contact.aspx
- Default.aspx
- favicon.ico
- Global.asax
- newHospitales.aspx
- packages.config
- SerchHospital.aspx
- SerchHospital.aspx.cs
- SerchHospital.aspx.designer.cs

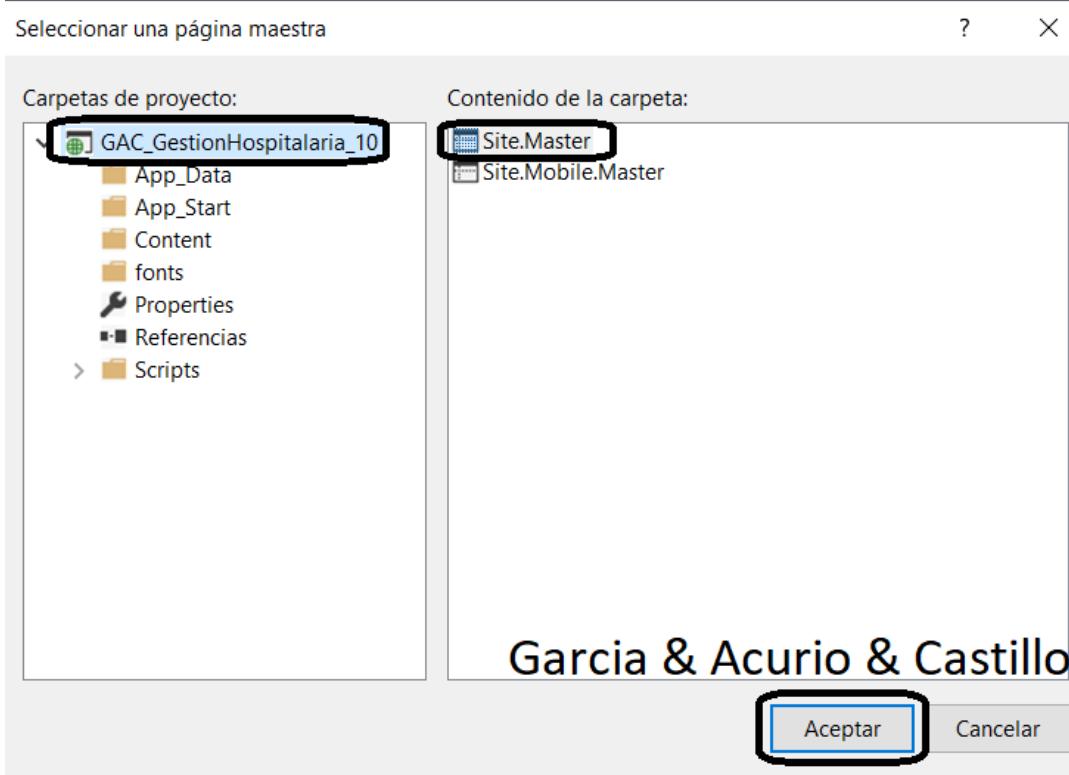
164. Luego ejecutamos y probamos los botones y nos debe redirigir a la página correspondientes

3.22. CODIFICACIÓN DEL DISEÑO ACTUALIZAR HOSPITAL

165. Hacemos clic derecho sobre la capa de presentación y **agregar- nuevo elemento**, elegimos formulario de tipo web con la opción de **página maestra**, colocamos el nombre de **updateHospital** y agregamos



166. Se nos mostrara una ventana, elegimos la opción **Site.Master** y luego aceptamos



167. Nos dirigimos al formulario **newHospitales.aspx** en la parte del **código** copiamos todo

The screenshot shows the Visual Studio IDE interface. The title bar indicates the project name "GAC_GestionHospitalaria_10". The main area is the code editor for the file "newHospitales.aspx", which contains ASPX code with C# script blocks. The code includes input fields for Código, Nombre, Tipo, and Categoría, along with dropdown lists for Tipo and Categoría. The "Explorador de soluciones" (Solution Explorer) pane on the right shows the project structure with files like Global.asax, Contact.aspx, Default.aspx, favicon.ico, and newHospitales.aspx. The "Propiedades" (Properties) pane is also visible on the far right. The code in the editor is highlighted with a blue border.

```
1 <%@ Page Title="" Language="C#" MasterPageFile="~/Site.Master" AutoEventWireup="true" CodeBehind="newHospitales.aspx.cs" Inherits="GAC_GestionHospitalaria_10.newHospitales" %>
2 <asp:Content ID="Content1" ContentPlaceHolderID="MainContent" runat="server">
3
4     <div>
5         <h2>Ingreso de Hospitales </h2>
6     </div>
7     <div class="row">
8         <div class="p-1">
9             <div class="input-group">
10                 <span class="input-group-addon" id="lbl_id" style="width:200px">Codigo:</span>
11                 <asp:TextBox ID="txt_Id_Hospital" runat="server" placeholder="Ingrese el Código" CssClass="form-control" Width="300px" />
12             </div>
13         </div>
14     </div>
15     <div class="row">
16         <div class="p-1">
17             <div class="input-group">
18                 <span class="input-group-addon" id="lbl_Nombre" style="width:200px">Nombre:</span>
19                 <asp:TextBox ID="txt_Nombre" runat="server" placeholder="Ingrese el Nombre" CssClass="form-control" Width="300px" />
20             </div>
21         </div>
22     </div>
23     <div class="row">
24         <div class="p-1">
25             <div class="input-group">
26                 <span class="input-group-addon" id="lbl_Tipo" style="width:200px">Tipo:</span>
27                 <asp:DropDownList ID="ddl_Tipo" runat="server" CssClass="form-control" AutoPostBack="True" Width="300px" />
28             </div>
29         </div>
30     </div>
31     <div class="row">
32         <div class="p-1">
33             <div class="input-group">
34                 <span class="input-group-addon" id="lbl_Categoría" style="width:200px">Categoria:</span>
35                 <asp:DropDownList ID="ddl_Categorías" runat="server" CssClass="form-control" AutoPostBack="True" Width="300px" />
36             </div>
37         </div>
38     </div>
39 </div>
40 </div>
```

168. Una vez copiado nos dirigimos al formulario **newHospital** y pegamos el contenido y modificamos el **título** y el nombre de los **botones**

```

1 <%@ Page Title="" Language="C#" MasterPageFile="~/Site.Master" AutoEventWireup="true" CodeBehind="updateHospital.aspx.cs" Inherits="GAC_GestionHospitalaria_10.updateHospital" %>
2 <asp:Content ID="Content1" ContentPlaceHolderID="MainContent" runat="server">
3     <div>
4         <h2>Actualizar Hospitales </h2>
5     </div>
6     <div class="row" >
7         <div class="p-1" >
8             <div class="input-group" >
9                 <span class="input-group-addon" id="lbl_id" style="width:200px" >Codigo:</span>
10                <asp:TextBox ID="txt_Id_Hospital" runat="server" placeholder="Ingrese el Código" CssClass="form-control" Width="300px" />
11            </div>
12        </div>
13        <div class="row" >
14            <div class="p-1" >
15                <div class="input-group" >
16                    <span class="input-group-addon" id="lbl_Nombre" style="width:200px" >Nombre:</span>
17                    <asp:TextBox ID="txt_Nombre" runat="server" placeholder="Ingrese el Nombre" CssClass="form-control" Width="300px" />
18                </div>
19            </div>
20        </div>
21    </div>
22    <div class="row" >
23        <div class="p-1" >
24            <div class="input-group" >
25                <span class="input-group-addon" id="lbl_Tipo" style="width:200px" >Tipo:</span>
26                <asp:DropDownList ID="ddl_Tipo" runat="server" CssClass="form-control" AutoPostBack="True" Width="300px" />
27            </div>
28        </div>
29    </div>
30    <div class="row" >
31        <div class="p-1" >
32            <div class="input-group" >
33                <span class="input-group-addon" id="lbl_Categoría" style="width:200px" >Categoria:</span>
34                <asp:DropDownList ID="ddl_Categorias" runat="server" CssClass="form-control" AutoPostBack="True" Width="300px" />
35            </div>
36        </div>
37    </div>
38    <div>
39    </div>
40</div>

```

169. Ejecutamos y nos deberá quedar como en la imagen

Gestión Hospitalarias Al Servicio de la Comunidad

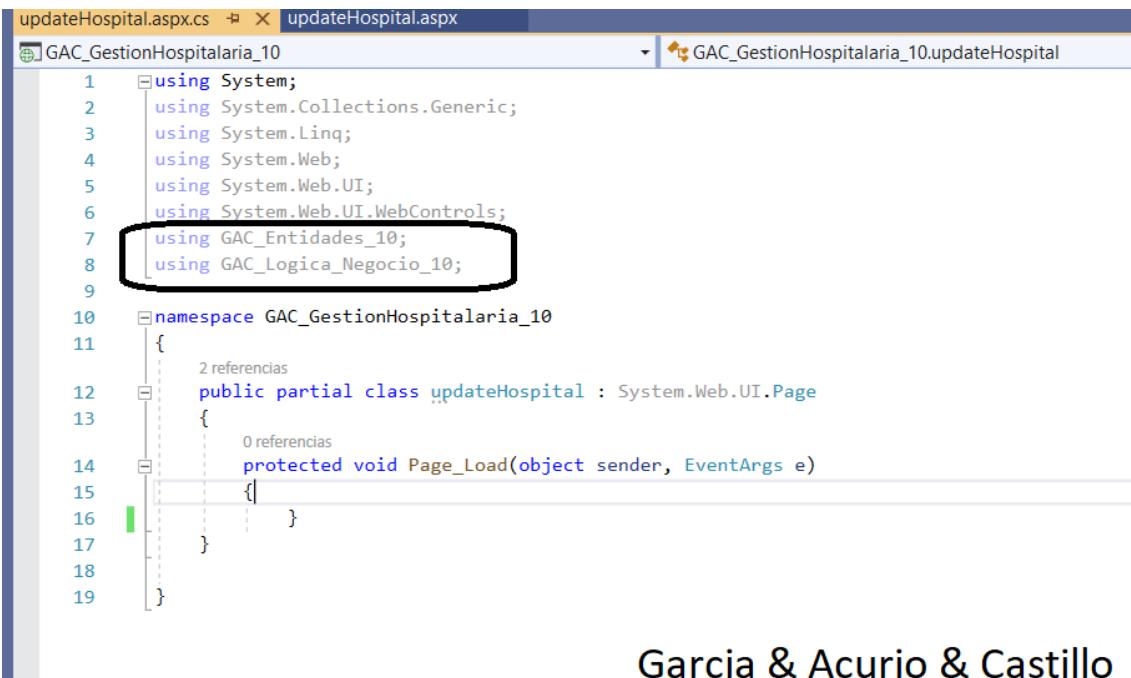
Código:	Ingrese el Código
Nombre:	Ingrese el Nombre
Tipo:	<input type="button"/>
Categoría:	<input type="button"/>
Área:	<input type="button"/>
Provincia:	<input type="button"/>
Cantón:	<input type="button"/>
Parroquia:	<input type="button"/>
Dirección:	Ingrese la Dirección
Fecha Creación:	Ingrese Fecha Creación
Gerente:	Ingrese el Gerente
Observación:	Ingrese la Observación

Actualizar **Salir**

Garcia & Acurio & Castillo

3.23. CODIFICACIÓN ACTUALIZAR HOSPITAL

170. Luego ponemos las librerías en updateHospital.aspx.cs



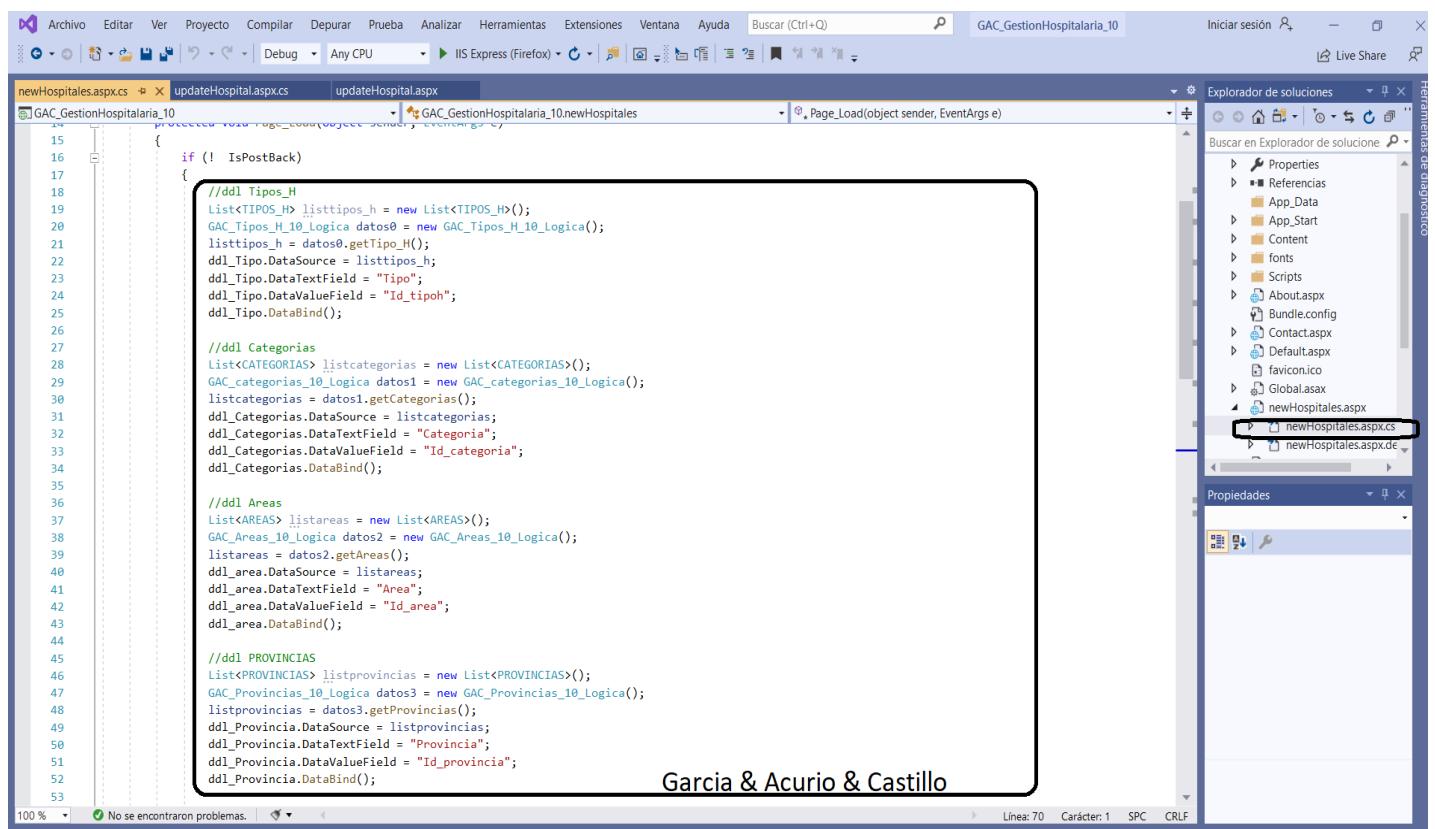
```
updateHospital.aspx.cs updateHospital.aspx
GAC_GestionHospitalaria_10 GAC_GestionHospitalaria_10.updateHospital

1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5  using System.Web.UI;
6  using System.Web.UI.WebControls;
7  using GAC_Entidades_10;
8  using GAC_Logica_Negocio_10;
9

10 namespace GAC_GestionHospitalaria_10
11 {
12     public partial class updateHospital : System.Web.UI.Page
13     {
14         protected void Page_Load(object sender, EventArgs e)
15         {
16         }
17     }
18 }
19 
```

Garcia & Acurio & Castillo

171. Luego nos dirigimos al newHospital.aspx.cs y copiamos el código como se muestra en la imagen



```
newHospitales.aspx updateHospital.aspx.cs updateHospital.aspx
GAC_GestionHospitalaria_10 GAC_GestionHospitalaria_10.newHospitales
newHospitales.aspx.cs

15 if (! IsPostBack)
16 {
17     //ddl Tipos_H
18     List<TIPOS_H> listtipos_h = new List<TIPOS_H>();
19     GAC_Tipos_H_10_Logica datos0 = new GAC_Tipos_H_10_Logica();
20     listtipos_h = datos0.getTipo_H();
21     ddl_Tipo.DataSource = listtipos_h;
22     ddl_Tipo.DataTextField = "Tipo";
23     ddl_Tipo.DataValueField = "Id_tipoh";
24     ddl_Tipo.DataBind();
25
26
27     //ddl Categorias
28     List<CATEGORIAS> listcategorias = new List<CATEGORIAS>();
29     GAC_categorias_10_Logica datos1 = new GAC_categorias_10_Logica();
30     listcategorias = datos1.getCategorias();
31     ddl_Categorias.DataSource = listcategorias;
32     ddl_Categorias.DataTextField = "Categoria";
33     ddl_Categorias.DataValueField = "Id_categoria";
34     ddl_Categorias.DataBind();
35
36
37     //ddl Areas
38     List<AREAS> listareas = new List<AREAS>();
39     GAC_Areas_10_Logica datos2 = new GAC_Areas_10_Logica();
40     listareas = datos2.getAreas();
41     ddl_area.DataSource = listareas;
42     ddl_area.DataTextField = "Area";
43     ddl_area.DataValueField = "Id_area";
44     ddl_area.DataBind();
45
46
47     //ddl PROVINCIAS
48     List<PROVINCIAS> listprovincias = new List<PROVINCIAS>();
49     GAC_Provincias_10_Logica datos3 = new GAC_Provincias_10_Logica();
50     listprovincias = datos3.getProvincias();
51     ddl_Provincia.DataSource = listprovincias;
52     ddl_Provincia.DataTextField = "Provincia";
53     ddl_Provincia.DataValueField = "Id_provincia";
54     ddl_Provincia.DataBind(); 
```

Garcia & Acurio & Castillo

172. Luego pegamos eso en el updateHospital.aspx.cs en el Page_Load

The screenshot shows the Visual Studio IDE. The code editor displays the C# code for the Page_Load event of the updateHospital.aspx page. The Solution Explorer on the right shows the project structure, including files like Contact.aspx, Default.aspx, Global.asax, newHospitales.aspx, and updateHospital.aspx. The Properties window is also visible.

```
newHospitales.aspx.cs
updateHospital.aspx.cs
updateHospital.aspx
GAC_GestionHospitalaria_10
GAC_GestionHospitalaria_10.updateHospital
Page_Load(object sender, EventArgs e)
{
    if (!IsPostBack)
    {
        //ddl Tipos_H
        List<TIPOS_H> listtipos_h = new List<TIPOS_H>();
        GAC_Tipos_H_10_Logica datos0 = new GAC_Tipos_H_10_Logica();
        listtipos_h = datos0.getTipos_H();
        ddl_Tipo.DataSource = listtipos_h;
        ddl_Tipo.DataTextField = "Tipo";
        ddl_Tipo.DataValueField = "Id_tipoh";
        ddl_Tipo.DataBind();

        //ddl Categorias
        List<CATEGORIAS> listcategorias = new List<CATEGORIAS>();
        GAC_Categorias_10_Logica datos1 = new GAC_Categorias_10_Logica();
        listcategorias = datos1.getCategorias();
        ddl_Categorias.DataSource = listcategorias;
        ddl_Categorias.DataTextField = "Categoria";
        ddl_Categorias.DataValueField = "Id_categoria";
        ddl_Categorias.DataBind();

        //ddl Areas
        List<AREAS> listareas = new List<AREAS>();
        GAC_Areas_10_Logica datos2 = new GAC_Areas_10_Logica();
        listareas = datos2.getAreas();
        ddl_area.DataSource = listareas;
        ddl_area.DataTextField = "Area";
        ddl_area.DataValueField = "Id_area";
        ddl_area.DataBind();

        //ddl PROVINCIAS
        List<PROVINCIAS> listprovincias = new List<PROVINCIAS>();
        GAC_Provincias_10_Logica datos3 = new GAC_Provincias_10_Logica();
        listprovincias = datos3.getProvincias();
        ddl_Provincia.DataSource = listprovincias;
        ddl_Provincia.DataTextField = "Provincia";
        ddl_Provincia.DataValueField = "Id_provincia";
    }
}
```

173. Ejecutamos y nos debe salir como en la imagen

Gestión Hospitalarias Al Servicio de la Comunidad

The screenshot shows a web application interface titled 'Actualizar Hospitales'. The form contains fields for Código, Nombre, Tipo, Categoría, Área, Provincia, Cantón, Parroquia, Dirección, and Fecha Creación. The Fecha Creación field is a date picker showing October 2020. Below the form are fields for Gerente and Observación, and buttons for Actualizar and Salir. The footer of the page displays the text 'Garcia & Acurio & Castillo'.

Código:	Ingrese el Código																																																				
Nombre:	Ingrese el Nombre																																																				
Tipo:	Hospital																																																				
Categoría:	A																																																				
Área:	Urbano Central																																																				
Provincia:	Esmerealdas																																																				
Cantón:	Esmerealdas																																																				
Parroquia:	Luis Tello																																																				
Dirección:	Ingrese la Dirección																																																				
Ingrese Fecha Creación																																																					
Fecha Creación:	<table border="1"><tr><td><</td><td>octubre de 2020</td><td>></td></tr><tr><td>lun.</td><td>mar.</td><td>mié.</td><td>jue.</td><td>vie.</td><td>sáb.</td><td>dom.</td></tr><tr><td>28</td><td>29</td><td>30</td><td>1</td><td>2</td><td>3</td><td>4</td></tr><tr><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>11</td></tr><tr><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td></tr><tr><td>19</td><td>20</td><td>21</td><td>22</td><td>23</td><td>24</td><td>25</td></tr><tr><td>26</td><td>27</td><td>28</td><td>29</td><td>30</td><td>31</td><td>1</td></tr><tr><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td></tr></table>	<	octubre de 2020	>	lun.	mar.	mié.	jue.	vie.	sáb.	dom.	28	29	30	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7	8
<	octubre de 2020	>																																																			
lun.	mar.	mié.	jue.	vie.	sáb.	dom.																																															
28	29	30	1	2	3	4																																															
5	6	7	8	9	10	11																																															
12	13	14	15	16	17	18																																															
19	20	21	22	23	24	25																																															
26	27	28	29	30	31	1																																															
2	3	4	5	6	7	8																																															
Gerente:	Ingrese el Gerente																																																				
Observación:	Ingrese la Observación																																																				
<input type="button" value="Actualizar"/>	<input type="button" value="Salir"/>																																																				

Garcia & Acurio & Castillo

174. Nos vamos al **Classroom** y copiamos el **código** que esta alojada hay



Jhonny Quiñonez

1 oct.

```
public HOSPITALES getOne(int registro_id)
{
    using (GestionHospitalariaEntities context = new GestionHospitalariaEntities())
    {
        var registros = from registro in context.HOSPITALES
                        where registro.Id_hospital == registro_id
                        select registro;
        return registros.First<HOSPITALES>();
    }
}
```

Garcia & Acurio & Castillo

175. Ir a la Capa “**GAC_Acceso_Datos_10**”, abrir la Clase “**GAC_Hospitales_10_Repository.cs**”, y pegar hay lo copiado del **Classroom** y modificar

```
59
60
61
62
63     // Listar Registro por ID
64
65     public List<HOSPITALES> getHospitales_id(int id_Hospital)
66     {
67         using (GestionHospitalariaEntities context = new GestionHospitalariaEntities())
68         {
69             var listhospital = from listhospital in context.HOSPITALES
70                             where listhospital.Id_hospital == id_Hospital
71                             select listhospital;
72             return listhospital.ToList();
73     }
74
75     public HOSPITALES get_Registro_Hospitales(int registro_id)
76     {
77         using (GestionHospitalariaEntities context = new GestionHospitalariaEntities())
78         {
79             var registros = from registro in context.HOSPITALES
80                             where registro.Id_hospital == registro_id
81                             select registro;
82             return registros.First<HOSPITALES>();
83     }
84
85 }
```

Garcia & Acurio & Castillo

176. Nos vamos al **Classroom** y copiamos el código que esta alojada hay



Jhonny Quiñonez

1 oct.

```
public HOSPITALES OtenerRegistro(int Registro_id)
{
    HOSPITALES listaregistros = new HOSPITALES();
    Hospitales_Repository datos = new Hospitales_Repository();
    listaregistros = datos.getOne(Registro_id);
    return listaregistros;
}
```

Garcia & Acurio & Castillo

177. Ir a la Capa “GAC_Logica_Negocio_10”, abrir la Clase “GAC_Hospitales_10_Logica.cs”, y pegar hay lo copiado del Classroom y modificar

```

42     GAC_Hospitales_10_Repository datos = new GAC_Hospitales_10_Repository();
43     listhosptiales = datos.getHospitales();
44     return listhosptiales;
45   }
46
47   //listar Registro por Id
48   public List<HOSPITALES> getHospitales_id(int id_Hospital)
49   {
50
51     List<HOSPITALES> listhosptiales = new List<HOSPITALES>();
52     GAC_Hospitales_10_Repository datos = new GAC_Hospitales_10_Repository();
53     listhosptiales = datos.getHospitales_id(id_Hospital);
54     return listhosptiales;
55   }
56
57   [referencia]
58   public HOSPITALES get_Registro_Hospital(int registro_id)
59   {
60     HOSPITALES listaregistros = new HOSPITALES();
61     GAC_Hospitales_10_Repository datos = new GAC_Hospitales_10_Repository();
62     listaregistros = datos.get_Registro_Hospitales(registro_id);
63     return listaregistros;
64   }
65
66 }

```

Garcia & Acurio & Castillo

178. Luego, codificamos el código como se muestra en la imagen hay mismo en el Page Load

```

71
72   int idhospital = Convert.ToInt32(Session["id_hospital"]);
73   GAC_Logica_Negocio_10.GAC_Hospitales_10_Logica datos6 = new GAC_Hospitales_10_Logica();
74   HOSPITALES registro_hospital = new HOSPITALES();
75   registro_hospital = datos6.get_Registro_Hospital(idhospital);
76
77
78   txt_Id_Hospital.Text = registro_hospital.Id_hospital.ToString();
79   txt_Nombre.Text = registro_hospital.Nombre.ToString();
80   ddl_Tipo.SelectedValue = registro_hospital.Tipo.ToString();
81   ddl_Categorias.SelectedValue = registro_hospital.Categorias.ToString();
82   ddl_Area.SelectedValue = registro_hospital.Area.ToString();
83   ddl_Provincia.SelectedValue = registro_hospital.Provincia.ToString();
84   ddl_Canton.SelectedValue = registro_hospital.Canton.ToString();
85   ddl_Parroquia.SelectedValue = registro_hospital.Parroquia.ToString();
86   txt_Direccion.Text = registro_hospital.Direccion.ToString();
87   txt_Fecha_Creacion.Text = registro_hospital.Fecha_Creacion.ToString();
88   txt_Gerente.Text = registro_hospital.Gerente.ToString();
89   txt_Observacion.Text = registro_hospital.Observacion.ToString();
90
91 }
92

```

Garcia & Acurio & Castillo

179. Ejecutamos y nos debe quedar como en la imagen

Gestión Hospitalarias Al Servicio de la Comunidad

Actualizar Hospitales

Codigo:	1																																																								
Nombre:	San Gregorio Loco																																																								
Tipo:	Hospital																																																								
Categoría:	A																																																								
Area:	Urbano Central																																																								
Provincia:	Esmerealdas																																																								
Canton:	Esmerealdas																																																								
Parroquia:	Luis Tello																																																								
Direccion:	Calle olmedo y Rocafuerte																																																								
Fecha Creacion:	15/10/2020 0:00:00 <table border="1" style="margin-left: auto; margin-right: auto;"><tr><td colspan="7">octubre de 2020</td></tr><tr><td><</td><td>lun.</td><td>mar.</td><td>mié.</td><td>jue.</td><td>vie.</td><td>sáb. dom.</td></tr><tr><td>28</td><td>29</td><td>30</td><td>1</td><td>2</td><td>3</td><td>4</td></tr><tr><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>11</td></tr><tr><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td></tr><tr><td>19</td><td>20</td><td>21</td><td>22</td><td>23</td><td>24</td><td>25</td></tr><tr><td>26</td><td>27</td><td>28</td><td>29</td><td>30</td><td>31</td><td>1</td></tr><tr><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td></tr></table>	octubre de 2020							<	lun.	mar.	mié.	jue.	vie.	sáb. dom.	28	29	30	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7	8
octubre de 2020																																																									
<	lun.	mar.	mié.	jue.	vie.	sáb. dom.																																																			
28	29	30	1	2	3	4																																																			
5	6	7	8	9	10	11																																																			
12	13	14	15	16	17	18																																																			
19	20	21	22	23	24	25																																																			
26	27	28	29	30	31	1																																																			
2	3	4	5	6	7	8																																																			
Gerente:	Dr. Luis Alba																																																								
Observacion:	Esta totalmente Renovado																																																								

[Actualizar](#) [Salir](#)

Garcia & Acurio & Castillo

180. Después damos Clic sobre el botón **Actualizar** y se nos abrirá el evento **onClick**, nos dirigimos al **newHospital.aspx.cs**, copiamos lo del botón **Guardar**

```
81 0 referencias
82
83
84 protected void btn_Guardar_Click(object sender, EventArgs e)
85 {
86
87     HOSPITALES nuevoHospital = new HOSPITALES();
88     nuevoHospital.Nombre = txt_Nombre.Text;
89     nuevoHospital.Tipo= Convert.ToInt32(ddl_Tipo.SelectedValue);
90     nuevoHospital.Categoría = Convert.ToInt32(ddl_Categorias.SelectedValue);
91     nuevoHospital.Area = Convert.ToInt32(ddl_area.SelectedValue);
92     nuevoHospital.Provincia = Convert.ToInt32(ddl_Provincia.SelectedValue);
93     nuevoHospital.Canton = Convert.ToInt32(ddl_Canton.SelectedValue);
94     nuevoHospital.Parroquia = Convert.ToInt32(ddl_Parroquia.SelectedValue);
95     nuevoHospital.Direccion = txt_Direccion.Text;
96     nuevoHospital.Fecha_Creacion = Convert.ToDateTime(txt_Fecha_Creacion.Text);
97     nuevoHospital.Gerente = txt_Gerente.Text;
98     nuevoHospital.Observacion = txt_Observacion.Text;
99
100 }

GAC_Hospitales_10_Logica datos = new GAC_Hospitales_10_Logica();
datos.NewHospital(nuevoHospital);
```

Garcia & Acurio & Castillo

181. Luego nos vamos al **updateHospital.aspx.cs** y pegamos en el evento **onclick** del botón **Actualizar**, modificamos

```
0 referencias
93 protected void btn_Actualizar_Click(object sender, EventArgs e)
94 {
95
96     HOSPITALES nuevoHospital = new HOSPITALES();
97     nuevoHospital.Id_hospital = Convert.ToInt32(txt_Id_Hospital.Text);
98     nuevoHospital.Nombre = txt_Nombre.Text;
99     nuevoHospital.Tipo = Convert.ToInt32(ddl_Tipo.SelectedValue);
100    nuevoHospital.Categoría = Convert.ToInt32(ddl_Categorias.SelectedValue);
101    nuevoHospital.Area = Convert.ToInt32(ddl_area.SelectedValue);
102    nuevoHospital.Provincia = Convert.ToInt32(ddl_Provincia.SelectedValue);
103    nuevoHospital.Canton = Convert.ToInt32(ddl_Canton.SelectedValue);
104    nuevoHospital.Parroquia = Convert.ToInt32(ddl_Parroquia.SelectedValue);
105    nuevoHospital.Direccion = txt_Direccion.Text;
106    nuevoHospital.Fecha_Creacion = Convert.ToDateTime(txt_Fecha_Creacion.Text);
107    nuevoHospital.Gerente = txt_Gerente.Text;
108    nuevoHospital.Observacion = txt_Observacion.Text;
109
110    GAC_Hospitales_10_Logica datos = new GAC_Hospitales_10_Logica();
111    datos.updateHospital(nuevoHospital);
```

Garcia & Acurio & Castillo

182. Luego actualizamos y debe guardar cambios

Administrar Hospitales

ID	Nombre	Tipo	Área	Categoría	Actualizar	Eliminar
1	San Gregorio Loco	2	3		Actualizar	Eliminar

[Nuevo](#) [Salir](#)

Garcia & Acurio & Castillo

Administrar Hospitales

ID	Nombre	Tipo	Área	Categoría	Actualizar	Eliminar
1	Clinica Santiago	1	1		Actualizar	Eliminar

[Nuevo](#) [Salir](#)

Garcia & Acurio & Castillo

183. Luego codificamos el botón salir

```
0 referencias
protected void btn_Salir_Click(object sender, EventArgs e)
{
    Response.Redirect("SerchHospital.aspx");
}
```

Garcia & Acurio & Castillo