Seminario II



**FACULTAD DE INGENIERÍA, ARQUITECTURA Y URBANISMO**

**Escuela Profesional de Ingeniería de Sistemas**

**SEMINARIO AVANZADO I**

**REGISTRO DE MATRICULA Y CONTROL DEL COLEGIO INSTITUCIONAL “ENRIQUE MILLA OCHOA”**

**TRABAJO INDIVIDUAL**

**Integrantes:**

Franco Rodríguez, Anthony Walter

**Docente:**

Atalaya Urrutia Carlos William

**INDICE**

[RESUMEN 3](#_Toc57513627)

[INTRODUCCIÓN 4](#_Toc57513628)

[REALIDAD PROBLEMÁTICA 5](#_Toc57513629)

[I. ASPECTOS GENERALES 6](#_Toc57513630)

[**1.1. DESCRIPCIÓN DE LA EMPRESA 6**](#_Toc57513631)

[**1.2. PROCESOS INVOLUCRADOS 9**](#_Toc57513632)

[**1.3. MARCO TEÓRICO 9**](#_Toc57513633)

[II. ANÁLISIS DE BASE DE DATOS 11](#_Toc57513634)

[2.1. DIAGRAMA DE BASE DE DATOS (DER) 11](#_Toc57513635)

[2.2. DESCRIPCIÓN DEL SCRIPT DE LA BASE DE DATOS 12](#_Toc57513636)

[III. DESARROLLO DE APLICACIÓN 25](#_Toc57513637)

[3.1. Descripción de la Aplicación: 25](#_Toc57513638)

[3.2. Diseño de Formularios (VISTAS) 25](#_Toc57513639)

[3.3. DISEÑO DE BASE DATOS 29](#_Toc57513640)

[3.4. Implementación de Code First y Entity framework 30](#_Toc57513641)

[CONCLUSIONES 34](#_Toc57513642)

[REFERENCIAS 35](#_Toc57513643)

[ANEXO 36](#_Toc57513644)

# **RESUMEN**

Se ha desarrollado un sistema en el lenguaje de programación .NET y como base de datos se ha utilizado SQL Server, este sistema servirá como herramienta que permite tener el control de las matrículas de los alumnos de la institución. Esta aplicación se ha desarrollado bajo los parámetros del curso *SEMINARIO AVANZADO I .*

# **INTRODUCCIÓN**

El presento proyecto está desarrollado utilizando .NET, la aplicación web tendrá responsivo para poder adaptarse a cualquier dispositivo este proyecto está basado en el lenguaje de programación C# , con ASP .NET.

Se construirán menús y formularios incluiros en páginas ASPX como parte de una aplicación web basados en hojas de estilos.

El desarrollo de la base de datos se realizará en el gestor SQL SERVER. Esta aplicación se desarrollará considerando la arquitectura de capas.

Los invitamos a revisar el contenido del presente trabajo, para poder apreciar todo el esfuerzo desarrollado y las técnicas de desarrollo utilizadas en todo el Proyecto.

# **REALIDAD PROBLEMÁTICA**

Actualmente la institución cuenta con procesos habituales de registro de alumnos y matriculas a un curso de manera tradicional.

Algunos procesos utilizan herramientas de ofimática para llevar un control de los registros de alumnos en los cursos y semestres indicados.

Esto lleva mucho tiempo para ordenar y organizar, además de sacar los reportes indicado por gerencia.

# **ASPECTOS GENERALES**

## **DESCRIPCIÓN DE LA EMPRESA**

La creación del Centro Educativo nació como una necesidad de la de los pobladores, que llegaron a instalarse al AA.HH en estas circunstancias por los años 1990, llagaron por primera vez la Profesora GOEORGINA VENTOSILLA AVILA, y se pone en contacto con los señores dirigentes de los pobladores; así, se inicia las gestiones educativas inscribiéndose a los niños y niñas en edad escolar, entonces nace la idea de la creación de un Centro Educativo Estatal, para atender la educación de los alumnos preparándose la documentación necesaria para dicha gestión; así al comienzo del mes de abril  del año de 1990, presenta la solicitud para la creación del Centro Educativo, a la USE 06 de San Martín de Porres  - Los Olivos; paralelamente se inicia las clases en ambientes provisionales estructurados (techos de esteras, mobiliario improvisados de madera y adobe), posteriormente a fines de junio del mismo año, se asigna en calidad de contratados a tres profesores.

Estructura Organizacional de la Empresa

**DIRECCIONAMIENTO ESTRATÉGICO EMPRESARIAL**

* **VISIÓN**

La Institución Educativa “Enrique Milla Ochoa” al 2016 será una Institución de calidad, que promueva el aprendizaje significativo con estudiantes competentes y con una infraestructura adecuada, considerando las nuevas exigencias de la sociedad moderna.

* **Misión**

“Somos una Entidad Educativa Estatal, cuya razón es formar alumnos eficaces en la práctica de valores sociales que contribuya al desarrollo de la persona en familia y de la comunidad”.

* **Principios y Valores**

El Colegio Institucional Educativa Enrique Milla Ochoa de Lima desarrolla una capacitación permanente en educación en valores, para formar personas profundamente humanas, que los practique con respeto, justicia, honestidad, solidaridad.

**Objetivos del Colegio**

**Objetivo General**

* Promover el aprendizaje integrando las TIC en las actividades pedagógicas, contribuyendo al desarrollo integral de estudiante y brindándoles las oportunidades en el marco de una sociedad del conocimiento.
* Crear en los docentes la necesidad y deseo de usar las XO 1.5, Incorporándolo como recurso importante en el proceso enseñanza-aprendizaje; para motivar, investigar, descubrir, imaginar y crear actividades pedagógicas.

**Objetivo Específicos**

* Capacitar a los docentes de la I.E. “ENRIQUE MILLA OCHOA” en el uso de recursos TIC. Para fortalecer el proceso de enseñanza y aprendizaje.
* Enseñar nociones básicas que le servirán a los docentes para el uso de las XO 1.5, para que apoyen a los alumnos con algunas actividades escolares.
* Promover en los alumnos el uso y aprovechamiento de las Tics para mejorar su aprendizaje y sus condiciones de vida.
* Organizar actividades económicas con la finalidad de generar recursos y adquirir computadoras con el apoyo de la comunidad educativa.

**Análisis Interno: Fortalezas y Debilidades**

**Fortalezas**

* El aula de innovación pedagógica está ubicada en un buen ambiente
* El Aula de Innovación Pedagógica cuenta con un docente a cargo.
* Capacitación de la DIGETE al docente encargado del aula de innovación pedagógica.
* Docentes predispuestos para desarrollar nuevas propuestas.
* Alumnos y docentes con acceso a las nuevas tecnologías en su casa o en las cabinas de internet

**Debilidades**

* Insuficientes computadoras para atender a todos los estudiantes
* Falta de mobiliario adecuado para las computadoras.
* No se cuenta con recursos económicos para el mejoramiento
* Falta de una cultura de programación de sesiones de aprendizaje
* Temor de los docentes al uso de la tecnología.

**Análisis Externo: Oportunidades y Amenazas**

**Oportunidades**

* La I. E. pertenece a una Red Educativa con los colegios cercanos de la institución
* Asesoramiento constante de la DIGETE.
* Apoyo de los padres de familia y docentes por apoyar el equipamiento del Aula.
* La I. E. participa en las capacitaciones que programa el Ministerio de Educación acerca de las Tics.
* Portales educativos donde se accede a todo tipo de información para realizar las sesiones.

**Amenazas**

* Indiferencia de la comunidad por el avance tecnológico.
* La resistencia al cambio producto de la educación tradicional.
* Constantes cambios de las normas emitidas por el MED y la UGEL.

## **PROCESOS INVOLUCRADOS**

Gestión de Usuarios

Se crea un administrador del dominio inicial cuando se crea un dominio nuevo. El administrador del dominio tiene los privilegios necesarios para gestionar el dominio desde el principio. El administrador del dominio puede utilizarse para crear y configurar usuarios, grupos, recursos y aplicaciones, así como para delegar tareas de administración del dominio, según sea necesario.

Gestión de Matricula

El proceso de matrícula forma parte de las actividades institucionales, se realiza preferentemente al finalizar el año escolar, está destinado a captar matrícula en todos los niveles del establecimiento y se mantiene abierto durante el resto del año.

## **MARCO TEÓRICO**

Definición del Proyecto

Nuestro objetivo principal del Proyecto es la generación de un Sistema Automatizado de Matricula (Registro de Cursos), en el cual los alumnos del Colegio Enrique Milla Ochoa, puedan realizar su inscripción en los diferentes cursos asignados según su carrera.

Esto permitirá poder tener un control exacto de los cursos a los cuales se registren los alumnos, para su posterior facturación.

Adicionalmente a esto podemos considerar como un objetivo, el desarrollo de un Sistema utilizando el patrón MVC para su programación.

Descripción del Módulo

Este módulo se encarga básicamente de la Matricula de Cursos de los alumnos del Colegio Enrique Milla Ochoa para el cual estos se validan mediante un Usuario (Código) y una contraseña, previamente creado, para que así puedan seleccionar sus cursos a llevar en el presente ciclo académico.

Adicionalmente a ello, permite la inserción de nuevos alumnos al sistema, así como la inserción de nuevos cursos al sistema, para la matricula correspondiente.

La importancia de utilizar MVC

La importancia de usa un patrón MVC depende de muchos factores, como recursos, tiempo, presupuesto e infraestructura.

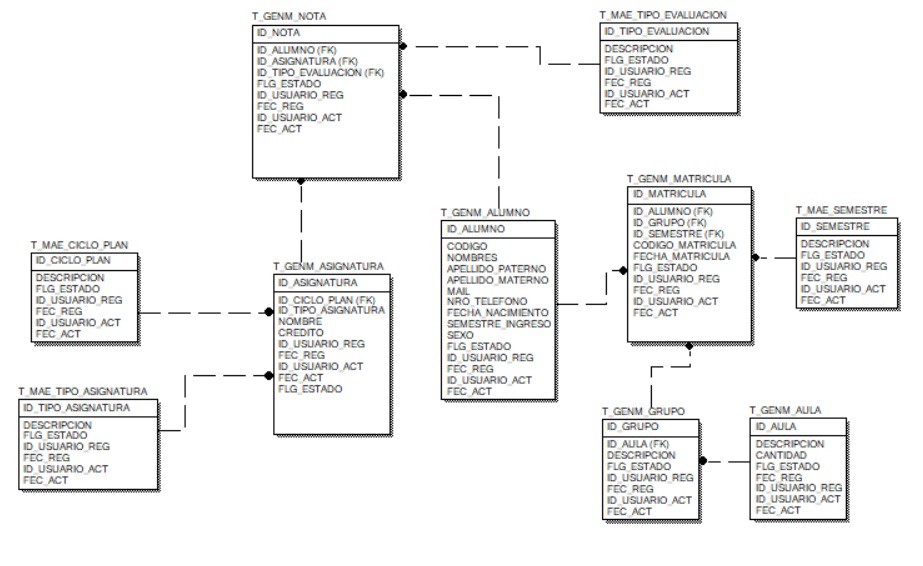
Porque, por un lado, tenemos que los patrones MVC son excelentes en el desarrollo pues ayudan a que el desarrollo sea más rápido, se reutiliza código, es muy bueno en solucionar problemas de desarrollo, el desarrollo de vuelve más escalable, ayuda a controlar la memoria cache y simplifica la documentación. Por otro lado, no siempre es necesario utilizarlo, pues usarlo o no usarlo no determinara la calidad del desarrollo, usarlo significa incrementar los recursos, la curva de aprendizaje es mayor y muchas veces los tiempos de desarrollo son cortos. (Mestras, 2008) (uniwebsidad, 2006)

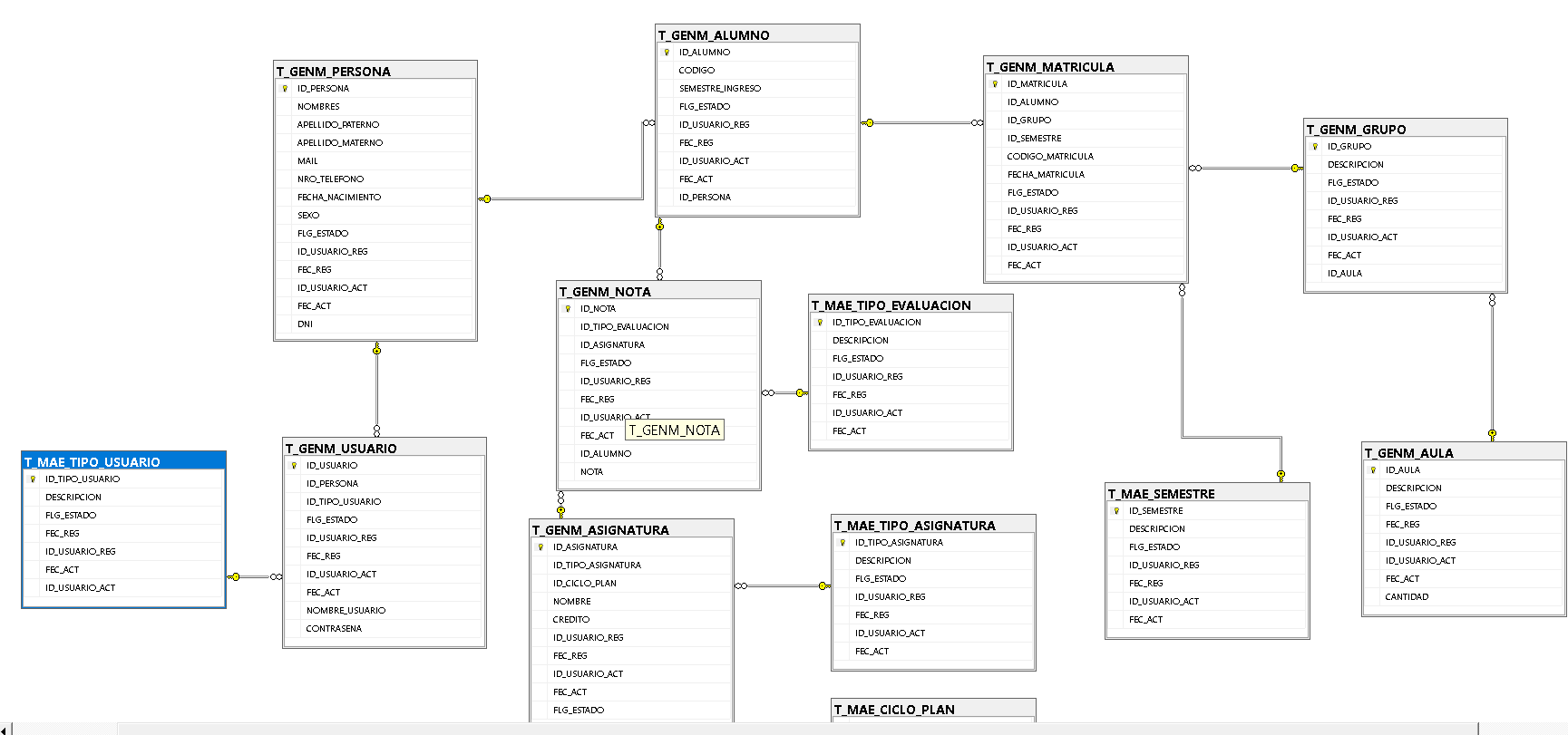
Alcance del Módulo

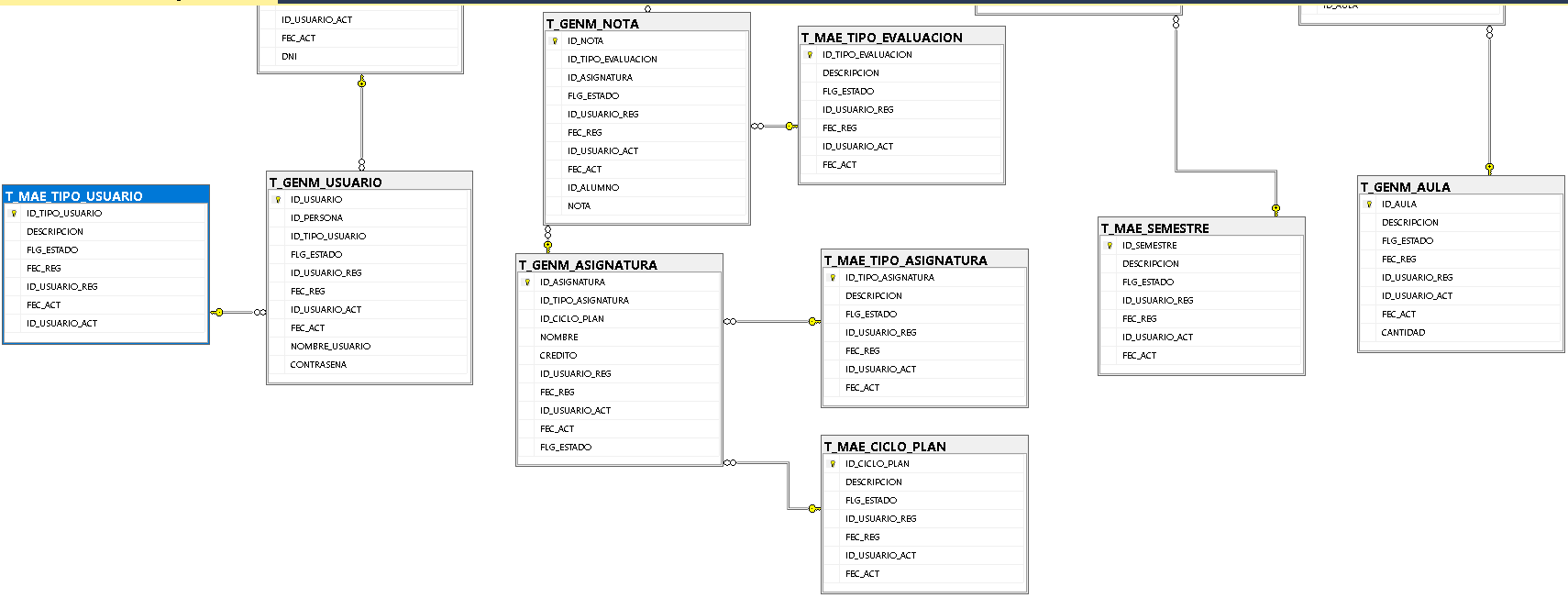
Este módulo tiene un alcance para el área Académica de Formación y es utilizado por todo el alumnado en general. Cabe indicar que, en la primera etapa, estará siendo utilizado únicamente por la secretaria académica, hasta su implementación final.

1. **ANÁLISIS DE BASE DE DATOS**
2. **DIAGRAMA DE BASE DE DATOS (DER)**

Diagrama Entidad Relación







1. **DESCRIPCIÓN DEL SCRIPT DE LA BASE DE DATOS**

**Script de Creación de Base de datos**

**/\*\*\* Crea la Base de datos en la unidad C\*\*\*/**

USE [master]

GO

**/\*\*\*\*\*\* Object: Database [BD\_MATRICULA] Script Date: 29/11/2020 00:40:45 \*\*\*\*\*\*/**

CREATE DATABASE [BD\_MATRICULA] ON PRIMARY

( NAME = N'BD\_MATRICULA', FILENAME = N'C:\Program Files\Microsoft SQL Server\MSSQL10\_50.MSSQLSERVER\MSSQL\DATA\BD\_MATRICULA.mdf' , SIZE = 3072KB , MAXSIZE = UNLIMITED, FILEGROWTH = 1024KB )

LOG ON

( NAME = N'BD\_MATRICULA\_log', FILENAME = N'C:\Program Files\Microsoft SQL Server\MSSQL10\_50.MSSQLSERVER\MSSQL\DATA\BD\_MATRICULA\_log.ldf' , SIZE = 1280KB , MAXSIZE = 2048GB , FILEGROWTH = 10%)

GO

USE [BD\_MATRICULA]

GO

**/\*\*\* Taba General de Alumnos\*\*\*/**

CREATE TABLE [dbo].[T\_GENM\_ALUMNO](

[ID\_ALUMNO] [int] IDENTITY(1,1) NOT NULL,

[CODIGO] [varchar](7) NULL,

[SEMESTRE\_INGRESO] [varchar](250) NULL,

[FLG\_ESTADO] [char](1) NULL,

[ID\_USUARIO\_REG] [int] NULL,

[FEC\_REG] [datetime] NULL,

[ID\_USUARIO\_ACT] [int] NULL,

[FEC\_ACT] [datetime] NULL,

[ID\_PERSONA] [int] NOT NULL,

CONSTRAINT [XPKT\_GENM\_ALUMNO] PRIMARY KEY CLUSTERED //Creación del índice

(

[ID\_ALUMNO] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

**/\*\*\*Tabla General de ASIGNATURA\*\*\*/**

GO

CREATE TABLE [dbo].[T\_GENM\_ASIGNATURA](

[ID\_ASIGNATURA] [int] IDENTITY(1,1) NOT NULL,

[ID\_TIPO\_ASIGNATURA] [int] NOT NULL,

[ID\_CICLO\_PLAN] [int] NOT NULL,

[NOMBRE] [varchar](250) NULL,

[CREDITO] [int] NULL,

[ID\_USUARIO\_REG] [int] NULL,

[FEC\_REG] [datetime] NULL,

[ID\_USUARIO\_ACT] [int] NULL,

[FEC\_ACT] [datetime] NULL,

[FLG\_ESTADO] [char](1) NULL,

CONSTRAINT [XPKT\_GENM\_CURSO] PRIMARY KEY CLUSTERED

(

[ID\_ASIGNATURA] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

**/\*\*\*Tabla General de Aula\*\*\*/**

GO

CREATE TABLE [dbo].[T\_GENM\_AULA](

[ID\_AULA] [int] IDENTITY(1,1) NOT NULL,

[DESCRIPCION] [varchar](250) NULL,

[FLG\_ESTADO] [char](1) NULL,

[FEC\_REG] [datetime] NULL,

[ID\_USUARIO\_REG] [int] NULL,

[ID\_USUARIO\_ACT] [int] NULL,

[FEC\_ACT] [datetime] NULL,

[CANTIDAD] [int] NULL,

CONSTRAINT [XPKT\_GENM\_AULA] PRIMARY KEY CLUSTERED

(

[ID\_AULA] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

**/\*\*\*Tabla General de GRUPO\*\*\*/**

GO

CREATE TABLE [dbo].[T\_GENM\_GRUPO](

[ID\_GRUPO] [int] IDENTITY(1,1) NOT NULL,

[DESCRIPCION] [varchar](250) NULL,

[FLG\_ESTADO] [char](1) NULL,

[ID\_USUARIO\_REG] [int] NULL,

[FEC\_REG] [datetime] NULL,

[ID\_USUARIO\_ACT] [int] NULL,

[FEC\_ACT] [datetime] NULL,

[ID\_AULA] [int] NOT NULL,

CONSTRAINT [XPKT\_GENM\_GRUPO] PRIMARY KEY CLUSTERED

(

[ID\_GRUPO] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

**/\*\*\*Tabla General de MATRICULA\*\*\*/**

GO

CREATE TABLE [dbo].[T\_GENM\_MATRICULA](

[ID\_MATRICULA] [int] IDENTITY(1,1) NOT NULL,

[ID\_ALUMNO] [int] NOT NULL,

[ID\_GRUPO] [int] NOT NULL,

[ID\_SEMESTRE] [int] NOT NULL,

[CODIGO\_MATRICULA] [varchar](7) NULL,

[FECHA\_MATRICULA] [datetime] NULL,

[FLG\_ESTADO] [char](1) NULL,

[ID\_USUARIO\_REG] [int] NULL,

[FEC\_REG] [datetime] NULL,

[ID\_USUARIO\_ACT] [int] NULL,

[FEC\_ACT] [datetime] NULL,

CONSTRAINT [XPKT\_GENM\_MATRICULA] PRIMARY KEY CLUSTERED

(

[ID\_MATRICULA] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

**/\*\*\*Tabla General de NOTA\*\*\*/**

GO

CREATE TABLE [dbo].[T\_GENM\_NOTA](

[ID\_NOTA] [int] IDENTITY(1,1) NOT NULL,

[ID\_TIPO\_EVALUACION] [int] NOT NULL,

[ID\_ASIGNATURA] [int] NOT NULL,

[FLG\_ESTADO] [char](1) NULL,

[ID\_USUARIO\_REG] [int] NULL,

[FEC\_REG] [datetime] NULL,

[ID\_USUARIO\_ACT] [int] NULL,

[FEC\_ACT] [datetime] NULL,

[ID\_ALUMNO] [int] NOT NULL,

[NOTA] [decimal](18, 2) NULL,

CONSTRAINT [XPKT\_GENM\_NOTA] PRIMARY KEY CLUSTERED

(

[ID\_NOTA] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

**/\*\*\*Tabla General de Persona\*\*\*/**

GO

CREATE TABLE [dbo].[T\_GENM\_PERSONA](

[ID\_PERSONA] [int] IDENTITY(1,1) NOT NULL,

[NOMBRES] [varchar](250) NULL,

[APELLIDO\_PATERNO] [varchar](250) NULL,

[APELLIDO\_MATERNO] [varchar](250) NULL,

[MAIL] [varchar](250) NULL,

[NRO\_TELEFONO] [varchar](9) NULL,

[FECHA\_NACIMIENTO] [datetime] NULL,

[SEXO] [char](1) NULL,

[FLG\_ESTADO] [char](1) NULL,

[ID\_USUARIO\_REG] [int] NULL,

[FEC\_REG] [datetime] NULL,

[ID\_USUARIO\_ACT] [int] NULL,

[FEC\_ACT] [datetime] NULL,

[DNI] [varchar](8) NULL,

CONSTRAINT [XPKT\_GENM\_PERSONA] PRIMARY KEY CLUSTERED

(

[ID\_PERSONA] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

**/\*\*\*Tabla General de Usuario\*\*\*/**

GO

CREATE TABLE [dbo].[T\_GENM\_USUARIO](

[ID\_USUARIO] [int] IDENTITY(1,1) NOT NULL,

[ID\_PERSONA] [int] NOT NULL,

[ID\_TIPO\_USUARIO] [int] NOT NULL,

[FLG\_ESTADO] [char](1) NULL,

[ID\_USUARIO\_REG] [int] NULL,

[FEC\_REG] [datetime] NULL,

[ID\_USUARIO\_ACT] [int] NULL,

[FEC\_ACT] [datetime] NULL,

[NOMBRE\_USUARIO] [varchar](50) NOT NULL,

[CONTRASENA] [varchar](50) NOT NULL,

CONSTRAINT [XPKT\_GENM\_USUARIO] PRIMARY KEY CLUSTERED

(

[ID\_USUARIO] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

**/\*\*\*Tabla Maestra CICLO PLAN\*\*\*/**

GO

CREATE TABLE [dbo].[T\_MAE\_CICLO\_PLAN](

[ID\_CICLO\_PLAN] [int] IDENTITY(1,1) NOT NULL,

[DESCRIPCION] [varchar](250) NULL,

[FLG\_ESTADO] [char](1) NULL,

[ID\_USUARIO\_REG] [int] NULL,

[FEC\_REG] [datetime] NULL,

[ID\_USUARIO\_ACT] [int] NULL,

[FEC\_ACT] [datetime] NULL,

CONSTRAINT [XPKT\_MAE\_CICLO\_PLAN] PRIMARY KEY CLUSTERED

(

[ID\_CICLO\_PLAN] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

**/\*\*\*Tabla Maestra SEMESTRE\*\*\*/**

GO

CREATE TABLE [dbo].[T\_MAE\_SEMESTRE](

[ID\_SEMESTRE] [int] IDENTITY(1,1) NOT NULL,

[DESCRIPCION] [varchar](250) NULL,

[FLG\_ESTADO] [char](1) NULL,

[ID\_USUARIO\_REG] [int] NULL,

[FEC\_REG] [datetime] NULL,

[ID\_USUARIO\_ACT] [int] NULL,

[FEC\_ACT] [datetime] NULL,

CONSTRAINT [XPKT\_MAE\_SEMESTRE] PRIMARY KEY CLUSTERED

(

[ID\_SEMESTRE] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

**/\*\*\*Tabla Maestra TIPO ASIGNATURA\*\*\*/**

GO

CREATE TABLE [dbo].[T\_MAE\_TIPO\_ASIGNATURA](

[ID\_TIPO\_ASIGNATURA] [int] IDENTITY(1,1) NOT NULL,

[DESCRIPCION] [varchar](250) NULL,

[FLG\_ESTADO] [char](1) NULL,

[ID\_USUARIO\_REG] [int] NULL,

[FEC\_REG] [datetime] NULL,

[ID\_USUARIO\_ACT] [int] NULL,

[FEC\_ACT] [datetime] NULL,

CONSTRAINT [XPKT\_MAE\_TIPO\_CURSO] PRIMARY KEY CLUSTERED

(

[ID\_TIPO\_ASIGNATURA] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

**/\*\*\*Tabla Maestra TIPO EVALUACION\*\*\*/**

GO

CREATE TABLE [dbo].[T\_MAE\_TIPO\_EVALUACION](

[ID\_TIPO\_EVALUACION] [int] IDENTITY(1,1) NOT NULL,

[DESCRIPCION] [varchar](250) NULL,

[FLG\_ESTADO] [char](1) NULL,

[ID\_USUARIO\_REG] [int] NULL,

[FEC\_REG] [datetime] NULL,

[ID\_USUARIO\_ACT] [int] NULL,

[FEC\_ACT] [datetime] NULL,

CONSTRAINT [XPKT\_MAE\_TIPO\_EVALUACION] PRIMARY KEY CLUSTERED

(

[ID\_TIPO\_EVALUACION] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

**/\*\*\*Tabla Maestra TIPO USUARIO\*\*\*/**

GO

CREATE TABLE [dbo].[T\_MAE\_TIPO\_USUARIO](

[ID\_TIPO\_USUARIO] [int] IDENTITY(1,1) NOT NULL,

[DESCRIPCION] [varchar](250) NULL,

[FLG\_ESTADO] [char](1) NULL,

[FEC\_REG] [datetime] NULL,

[ID\_USUARIO\_REG] [int] NULL,

[FEC\_ACT] [datetime] NULL,

[ID\_USUARIO\_ACT] [int] NULL,

CONSTRAINT [XPKT\_MAE\_TIPO\_USUARIO] PRIMARY KEY CLUSTERED

(

[ID\_TIPO\_USUARIO] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

**Nota: se ha creado unos esquemas de tablas en donde las Tablas GEN son tables generales que contiene relación con otras tablas, Tablas MAE con tablas Maestras que no contiene mantenimiento.**

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[PRD\_INSERTAR\_MATRICULA] Script Date: 29/11/2020 00:40:46 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[PRD\_INSERTAR\_MATRICULA](

@IN\_NOMBRES VARCHAR(250)

,@IN\_APELLIDO\_PATERNO VARCHAR(250)

,@IN\_APELLIDO\_MATERNO VARCHAR(250)

,@IN\_MAIL VARCHAR(250)

,@IN\_NRO\_TELEFONO VARCHAR(250)

,@IN\_FECHA\_NACIMIENTO DATE

,@IN\_SEMESTRE\_INGRESO VARCHAR(250)

,@IN\_SEXO CHAR(1)

,@IN\_ID\_GRUPO INTEGER

,@IN\_ID\_SEMESTRE INTEGER

,@IN\_FECHA\_MATRICULA DATE

,@IN\_ID\_USUARIO\_REG INTEGER

,@IN\_ID\_AULA INTEGER

,@DNI VARCHAR(8)

)

AS

BEGIN

DECLARE @Alumno\_Id INTEGER;

DECLARE @contador INTEGER;

DECLARE @cantidad INTEGER;

DECLARE @IN\_CODIGO\_ALUMNO VARCHAR(7);

DECLARE @IN\_CODIGO\_MATRICULA VARCHAR(7);

DECLARE @Persona\_Id INTEGER;

SELECT @cantidad=cantidad

FROM

T\_GENM\_GRUPO G

INNER JOIN T\_GENM\_AULA A ON G.ID\_AULA = A.ID\_AULA

WHERE G.ID\_GRUPO = @IN\_ID\_SEMESTRE AND A.ID\_AULA = @IN\_ID\_AULA AND G.FLG\_ESTADO = '1' AND A.FLG\_ESTADO = '1';

SELECT @contador=COUNT (1)

FROM

T\_GENM\_MATRICULA

WHERE ID\_SEMESTRE = @IN\_ID\_SEMESTRE AND ID\_GRUPO = @IN\_ID\_AULA AND FLG\_ESTADO = '1';

IF @cantidad = @contador

BEGIN

SELECT '0' AS RESULTADO,'La aula selecciona ya no cuenta con cupos. Intente matricularse en otra aula' AS MENSAJE;

END

ELSE

BEGIN

BEGIN TRANSACTION

BEGIN TRY

SET @IN\_CODIGO\_ALUMNO = (SELECT RIGHT('0000000' + Ltrim(Rtrim(CAST(ISNULL(MAX(CODIGO),'0') AS INTEGER) + 1)),7)

FROM T\_GENM\_ALUMNO);

SELECT @contador=COUNT (1)

FROM T\_GENM\_PERSONA

WHERE LTRIM(RTRIM(DNI)) = LTRIM(RTRIM(@DNI));

IF @contador = 0

BEGIN

INSERT INTO dbo.T\_GENM\_PERSONA

(NOMBRES, APELLIDO\_PATERNO, APELLIDO\_MATERNO, MAIL, NRO\_TELEFONO, FECHA\_NACIMIENTO

,SEXO, FLG\_ESTADO, ID\_USUARIO\_REG, FEC\_REG, DNI)

VALUES

(@IN\_NOMBRES, @IN\_APELLIDO\_PATERNO, @IN\_APELLIDO\_MATERNO, @IN\_MAIL, @IN\_NRO\_TELEFONO, @IN\_FECHA\_NACIMIENTO

,@IN\_SEXO, '1', @IN\_ID\_USUARIO\_REG, GETDATE(), @DNI);

SET @Persona\_Id = @@IDENTITY;

END

ELSE

BEGIN

SET @Persona\_Id = ( SELECT ID\_PERSONA FROM T\_GENM\_PERSONA WHERE LTRIM(RTRIM(DNI)) = LTRIM(RTRIM(@DNI)))

END

INSERT INTO dbo.T\_GENM\_ALUMNO

(CODIGO , SEMESTRE\_INGRESO, FLG\_ESTADO, ID\_USUARIO\_REG, FEC\_REG, ID\_PERSONA)

VALUES

(@IN\_CODIGO\_ALUMNO, @IN\_SEMESTRE\_INGRESO, '1', @IN\_ID\_USUARIO\_REG, GETDATE(), @Persona\_Id);

SET @Alumno\_Id = @@IDENTITY;

SET @IN\_CODIGO\_MATRICULA = (SELECT RIGHT('0000000' + Ltrim(Rtrim(CAST(ISNULL(MAX(CODIGO\_MATRICULA),'0') AS INTEGER) + 1)),7)

FROM T\_GENM\_MATRICULA);

INSERT INTO dbo.T\_GENM\_MATRICULA

(ID\_ALUMNO, ID\_GRUPO, ID\_SEMESTRE, CODIGO\_MATRICULA, FECHA\_MATRICULA, FLG\_ESTADO, ID\_USUARIO\_REG, FEC\_REG)

VALUES

(@Alumno\_Id, @IN\_ID\_GRUPO, @IN\_ID\_SEMESTRE, @IN\_CODIGO\_MATRICULA, @IN\_FECHA\_MATRICULA, '1', @IN\_ID\_USUARIO\_REG, GETDATE())

COMMIT TRANSACTION

SELECT '1' AS RESULTADO,'Se registro correctamente' AS MENSAJE;

END TRY

BEGIN CATCH

ROLLBACK TRANSACTION

SELECT 0 AS IDUSUARIO, '0' AS RESULTADO,'Error al procesar la transacción' AS MENSAJE;

END CATCH

END

END

/\*

EXEC PRD\_INSERTAR\_MATRICULA @IN\_NOMBRES = 'MARWGORI PENA', @IN\_APELLIDO\_PATERNO = 'CACERESD', @IN\_APELLIDO\_MATERNO = 'RUDIZ'

,@IN\_MAIL = 'margoriyua@gmail.com', @IN\_NRO\_TELEFONO = '951075457', @IN\_FECHA\_NACIMIENTO = '01/01/1990'

,@IN\_SEMESTRE\_INGRESO = '2020 - I', @IN\_SEXO = 'F', @IN\_ID\_GRUPO = 1, @IN\_ID\_SEMESTRE = 1, @IN\_FECHA\_MATRICULA = '07/11/2020'

,@IN\_ID\_USUARIO\_REG = 1, @IN\_ID\_AULA = 2, @DNI = '8045521'

\*/

**Nota:** Este procedimiento te valida la cantidad de vacante por aula si no hay cantidad retorna un mensaje: **'El aula selecciona ya no cuenta con cupos. Intente matricularse en otra aula'**

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[PRD\_INSERTAR\_PERSONA\_USUARIO] Script Date: 29/11/2020 00:40:46 \*\*\*\*\*\*/

GO

CREATE PROCEDURE [dbo].[PRD\_INSERTAR\_PERSONA\_USUARIO](

@IN\_NOMBRES VARCHAR(250)

,@IN\_APELLIDO\_PATERNO VARCHAR(250)

,@IN\_APELLIDO\_MATERNO VARCHAR(250)

,@IN\_MAIL VARCHAR(250)

,@IN\_NRO\_TELEFONO VARCHAR(250)

,@IN\_FECHA\_NACIMIENTO DATE

,@IN\_SEXO CHAR(1)

,@IN\_ID\_USUARIO\_REG INTEGER

,@DNI VARCHAR(8)

,@IN\_ID\_TIPO\_USUARIO INTEGER

,@IN\_NOMBRE\_USUARIO VARCHAR(50)

,@IN\_CONTRASENA VARCHAR(50)

)

AS

BEGIN

DECLARE @contador INTEGER;

DECLARE @Persona\_Id INTEGER;

BEGIN TRANSACTION

BEGIN TRY

SELECT @contador=COUNT (1)

FROM T\_GENM\_PERSONA

WHERE LTRIM(RTRIM(DNI)) = LTRIM(RTRIM(@DNI));

IF @contador = 0

BEGIN

INSERT INTO dbo.T\_GENM\_PERSONA

(NOMBRES, APELLIDO\_PATERNO, APELLIDO\_MATERNO, MAIL, NRO\_TELEFONO, FECHA\_NACIMIENTO

,SEXO, FLG\_ESTADO, ID\_USUARIO\_REG, FEC\_REG, DNI)

VALUES

(@IN\_NOMBRES, @IN\_APELLIDO\_PATERNO, @IN\_APELLIDO\_MATERNO, @IN\_MAIL, @IN\_NRO\_TELEFONO, @IN\_FECHA\_NACIMIENTO

,@IN\_SEXO, '1', @IN\_ID\_USUARIO\_REG, GETDATE(), @DNI);

SET @Persona\_Id = @@IDENTITY;

END

ELSE

BEGIN

SET @Persona\_Id = ( SELECT ID\_PERSONA FROM T\_GENM\_PERSONA WHERE LTRIM(RTRIM(DNI)) = LTRIM(RTRIM(@DNI)))

END

SELECT @contador=COUNT (1)

FROM

T\_GENM\_PERSONA P

INNER JOIN T\_GENM\_USUARIO U ON P.ID\_PERSONA = U.ID\_PERSONA

WHERE LTRIM(RTRIM(P.DNI)) = LTRIM(RTRIM(@DNI));

IF @contador = 0

BEGIN

INSERT INTO T\_GENM\_USUARIO

(ID\_PERSONA, ID\_TIPO\_USUARIO, FLG\_ESTADO, ID\_USUARIO\_REG

,FEC\_REG, NOMBRE\_USUARIO, CONTRASENA)

VALUES

(@Persona\_Id, @IN\_ID\_TIPO\_USUARIO, '1', @IN\_ID\_USUARIO\_REG

,GETDATE(), @IN\_NOMBRE\_USUARIO, @IN\_CONTRASENA)

END

ELSE

BEGIN

UPDATE T\_GENM\_USUARIO

SET ID\_TIPO\_USUARIO = @IN\_ID\_TIPO\_USUARIO,

FLG\_ESTADO = '1',

ID\_USUARIO\_ACT = @IN\_ID\_USUARIO\_REG,

FEC\_ACT = GETDATE(),

NOMBRE\_USUARIO = @IN\_NOMBRE\_USUARIO,

CONTRASENA = @IN\_CONTRASENA

WHERE ID\_PERSONA IN (SELECT ID\_PERSONA FROM T\_GENM\_PERSONA WHERE LTRIM(RTRIM(DNI)) = LTRIM(RTRIM(@DNI)))

END

COMMIT TRANSACTION

SELECT '1' AS RESULTADO,'Se registro correctamente' AS MENSAJE;

END TRY

BEGIN CATCH

ROLLBACK TRANSACTION

SELECT 0 AS IDUSUARIO, '0' AS RESULTADO,'Error al procesar la transacción' AS MENSAJE;

END CATCH

END

/\*

EXEC PRD\_INSERTAR\_PERSONA\_USUARIO @IN\_NOMBRES = 'MARWGORI PENA', @IN\_APELLIDO\_PATERNO = 'CACERESD', @IN\_APELLIDO\_MATERNO = 'RUDIZ'

,@IN\_MAIL = 'margoriyua@gmail.com', @IN\_NRO\_TELEFONO = '951075457', @IN\_FECHA\_NACIMIENTO = '01/01/1990'

,@IN\_SEXO = 'F', @IN\_ID\_USUARIO\_REG = 1, @DNI = '8045521', @IN\_ID\_TIPO\_USUARIO = 1

,@IN\_NOMBRE\_USUARIO ='CUSER1', @IN\_CONTRASENA = '123456'

\*/

**Nota: Este procedimiento crea una persona y a la vez asigna un usuario para el ingreso al sistema.**

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[PRD\_LISTAR\_ALUMNO] Script Date: 29/11/2020 00:40:46 \*\*\*\*\*\*/

GO

CREATE PROCEDURE [dbo].[PRD\_LISTAR\_ALUMNO]

AS

BEGIN

SELECT DISTINCT A.ID\_ALUMNO

,A.NOMBRES + ' ' + A.APELLIDO\_PATERNO + ' ' + A.APELLIDO\_MATERNO AS NOMBRES

FROM T\_GENM\_ALUMNO A

WHERE A.FLG\_ESTADO = '1' ;

END;

**Nota:** Este procedimiento lista los alumnos matriculado

--EXEC PRD\_LISTAR\_PARTES\_VEH

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[PRD\_LISTAR\_ALUMNO\_AULA] Script Date: 29/11/2020 00:40:46 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[PRD\_LISTAR\_ALUMNO\_AULA](

@IN\_ID\_GRUPO INTEGER,

@IN\_ID\_AULA INTEGER,

@IN\_ID\_SEMESTRE INTEGER

)

AS

BEGIN

SELECT A.ID\_ALUMNO

,A.CODIGO

,P.NOMBRES+' '+P.APELLIDO\_PATERNO+' '+P.APELLIDO\_MATERNO as NOMBRES

,P.APELLIDO\_PATERNO

,P.APELLIDO\_MATERNO

,P.MAIL

,P.NRO\_TELEFONO

,P.FECHA\_NACIMIENTO

,P.SEXO

FROM T\_GENM\_ALUMNO A

INNER JOIN T\_GENM\_PERSONA P ON A.ID\_PERSONA=P.ID\_PERSONA

INNER JOIN T\_GENM\_MATRICULA M ON A.ID\_ALUMNO = M.ID\_ALUMNO

INNER JOIN T\_GENM\_GRUPO G ON M.ID\_GRUPO = G.ID\_GRUPO

INNER JOIN T\_GENM\_AULA AL ON G.ID\_AULA = AL.ID\_AULA

WHERE A.FLG\_ESTADO = '1' AND M.FLG\_ESTADO = '1' AND G.FLG\_ESTADO = '1' AND AL.FLG\_ESTADO = '1'

AND G.ID\_GRUPO = @IN\_ID\_GRUPO AND AL.ID\_AULA = @IN\_ID\_AULA AND M.ID\_SEMESTRE = @IN\_ID\_SEMESTRE;

END;

**Nota:** Este procedimiento lista los alumnos por aula

--EXEC PRD\_LISTAR\_PARTES\_VEH

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[PRD\_LISTAR\_ALUMNO\_MATRICULA] Script Date: 29/11/2020 00:40:46 \*\*\*\*\*\*/

GO

CREATE PROCEDURE [dbo].[PRD\_LISTAR\_ALUMNO\_MATRICULA](

@IN\_NOMBRE\_COMPLETO VARCHAR(250),

@IN\_ID\_SEMESTRE INTEGER

)

AS

BEGIN

SELECT MAT.CODIGO\_MATRICULA, P.NOMBRES + ' ' + P.APELLIDO\_PATERNO + ' ' + P.APELLIDO\_MATERNO AS NOMBRES,

ASI.NOMBRE AS ASIGNATURA, TE.DESCRIPCION AS TIPO\_ASIGNATURA,

N.NOTA, S.DESCRIPCION AS SEMESTRE, C.DESCRIPCION AS CICLO

FROM T\_GENM\_ALUMNO A

INNER JOIN T\_GENM\_PERSONA P ON A.ID\_PERSONA=P.ID\_PERSONA

INNER JOIN T\_GENM\_NOTA N ON A.ID\_ALUMNO = N.ID\_ALUMNO

INNER JOIN T\_MAE\_TIPO\_EVALUACION TE ON N.ID\_TIPO\_EVALUACION = TE.ID\_TIPO\_EVALUACION

INNER JOIN T\_GENM\_ASIGNATURA ASI ON N.ID\_ASIGNATURA = ASI.ID\_ASIGNATURA

INNER JOIN T\_GENM\_MATRICULA MAT ON A.ID\_ALUMNO = MAT.ID\_ALUMNO

INNER JOIN T\_MAE\_SEMESTRE S ON MAT.ID\_SEMESTRE = S.ID\_SEMESTRE

INNER JOIN T\_MAE\_CICLO\_PLAN C ON ASI.ID\_CICLO\_PLAN = C.ID\_CICLO\_PLAN

WHERE A.ID\_ALUMNO=@IN\_NOMBRE\_COMPLETO

AND S.ID\_SEMESTRE = @IN\_ID\_SEMESTRE AND MAT.FLG\_ESTADO = '1' AND N.FLG\_ESTADO = '1' AND ASI.FLG\_ESTADO = '1'

END;

--EXEC PRD\_LISTAR\_ALUMNO\_MATRICULA @IN\_NOMBRE\_COMPLETO = 'DI', @IN\_ID\_SEMESTRE = 1

**Nota:** Este procedimiento lista los alumnos matriculado

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[PRD\_LISTAR\_ASIGNATURAS] Script Date: 29/11/2020 00:40:46 \*\*\*\*\*\*/

GO

CREATE PROCEDURE [dbo].[PRD\_LISTAR\_ASIGNATURAS]

AS

BEGIN

-- routine body goes here, e.g.

-- SELECT 'Navicat for SQL Server'

SELECT ID\_ASIGNATURA,NOMBRE FROM T\_GENM\_ASIGNATURA WHERE FLG\_ESTADO=1;

END

**Nota:** Este procedimiento lista la asignatura

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[PRD\_LISTAR\_AULA] Script Date: 29/11/2020 00:40:46 \*\*\*\*\*\*/

GO

CREATE PROCEDURE [dbo].[PRD\_LISTAR\_AULA]

(

@IN\_ID\_GRUPO INTEGER

)

AS

BEGIN

SELECT A.ID\_GRUPO

,A.DESCRIPCION AS GRUPO

,AL.ID\_AULA

,AL.DESCRIPCION AS AULA

FROM T\_GENM\_GRUPO A

INNER JOIN T\_GENM\_AULA AL ON A.ID\_AULA = AL.ID\_AULA

WHERE A.FLG\_ESTADO = '1' AND AL.FLG\_ESTADO = '1'

AND A.ID\_GRUPO = @IN\_ID\_GRUPO;

END;

--EXEC PRD\_LISTAR\_AULA @IN\_ID\_GRUPO = 1

**Nota:** Este procedimiento lista las aulas por grupo

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[PRD\_LISTAR\_AULAS] Script Date: 29/11/2020 00:40:46 \*\*\*\*\*\*/

GO

CREATE PROCEDURE [dbo].[PRD\_LISTAR\_AULAS]

AS

BEGIN

-- routine body goes here, e.g.

-- SELECT 'Navicat for SQL Server'

SELECT ID\_AULA ,DESCRIPCION FROM T\_GENM\_AULA WHERE FLG\_ESTADO=1

END

**Nota:** Este procedimiento lista las aulas

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[PRD\_LISTAR\_CURSO] Script Date: 29/11/2020 00:40:46 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[PRD\_LISTAR\_CURSO]

AS

BEGIN

SELECT A.ID\_ASIGNATURA

,A.NOMBRE

FROM T\_GENM\_ASIGNATURA A

WHERE A.FLG\_ESTADO = '1' ;

END;

--EXEC PRD\_LISTAR\_PARTES\_VEH

**Nota:** Este procedimiento lista los cursos

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[PRD\_LISTAR\_GRUPO] Script Date: 29/11/2020 00:40:46 \*\*\*\*\*\*/

GO

CREATE PROCEDURE [dbo].[PRD\_LISTAR\_GRUPO]

AS

BEGIN

SELECT A.ID\_GRUPO

,A.DESCRIPCION

FROM T\_GENM\_GRUPO A

WHERE A.FLG\_ESTADO = '1' ;

END;

--EXEC PRD\_LISTAR\_GRUPO

**Nota:** Este procedimiento lista los grupos

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[PRD\_LISTAR\_ORDEN\_MERITO] Script Date: 29/11/2020 00:40:46 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[PRD\_LISTAR\_ORDEN\_MERITO](

@IN\_ID\_ASIGNATURA INTEGER,

@IN\_ID\_GRUPO INTEGER,

@IN\_ID\_AULA INTEGER,

@IN\_ID\_SEMESTRE INTEGER

)

AS

BEGIN

SELECT ALIAS.\*

FROM

(

SELECT

DENSE\_RANK() OVER(order by X.NOMBRES, X.NOTA\_TOTAL DESC) AS ROWNUM,

X.NOMBRES, X.TIPO\_ASIGNATURA, X.NOTA, X.NOTA\_TOTAL

FROM

(

SELECT P.APELLIDO\_PATERNO + ' ' + P.APELLIDO\_MATERNO + ',' + P.NOMBRES AS NOMBRES, TE.DESCRIPCION AS TIPO\_ASIGNATURA,

N.NOTA,--G.DESCRIPCION AS GRUPO, AL.DESCRIPCION AS AULA

(SELECT SUM(NOTA) FROM T\_GENM\_NOTA WHERE ID\_ALUMNO = A.ID\_ALUMNO AND ID\_ASIGNATURA = ASI.ID\_ASIGNATURA) AS NOTA\_TOTAL

FROM T\_GENM\_ALUMNO A

INNER JOIN T\_GENM\_PERSONA P ON A.ID\_PERSONA=P.ID\_PERSONA

INNER JOIN T\_GENM\_NOTA N ON A.ID\_ALUMNO = N.ID\_ALUMNO

INNER JOIN T\_MAE\_TIPO\_EVALUACION TE ON N.ID\_TIPO\_EVALUACION = TE.ID\_TIPO\_EVALUACION

INNER JOIN T\_GENM\_ASIGNATURA ASI ON N.ID\_ASIGNATURA = ASI.ID\_ASIGNATURA

INNER JOIN T\_GENM\_MATRICULA MAT ON A.ID\_ALUMNO = MAT.ID\_ALUMNO

INNER JOIN T\_GENM\_GRUPO G ON MAT.ID\_GRUPO = G.ID\_GRUPO

INNER JOIN T\_GENM\_AULA AL ON G.ID\_AULA = AL.ID\_AULA

WHERE ASI.ID\_ASIGNATURA = @IN\_ID\_ASIGNATURA AND MAT.ID\_SEMESTRE = @IN\_ID\_SEMESTRE

AND ( 0 = @IN\_ID\_GRUPO OR G.ID\_GRUPO = @IN\_ID\_GRUPO)

AND ( 0 = @IN\_ID\_AULA OR AL.ID\_AULA = @IN\_ID\_AULA)

AND MAT.FLG\_ESTADO = '1' AND N.FLG\_ESTADO = '1'

) X

) ALIAS

WHERE ALIAS.ROWNUM <= 3

END;

--EXEC PRD\_LISTAR\_ORDEN\_MERITO @IN\_ID\_ASIGNATURA = 1, @IN\_ID\_GRUPO = 1, @IN\_ID\_AULA = 0, @IN\_ID\_SEMESTRE = 1

**Nota: evalúa en una sub consulta la mayor nota de la asignatura y a su vez retorna el tipo de evaluación, mostrando las 3 notas correspondiente de cada asignatura.**

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[PRD\_LISTAR\_SEMESTRE] Script Date: 29/11/2020 00:40:46 \*\*\*\*\*\*/

GO

CREATE PROCEDURE [dbo].[PRD\_LISTAR\_SEMESTRE]

AS

BEGIN

SELECT ID\_SEMESTRE,DESCRIPCION FROM T\_MAE\_SEMESTRE WHERE FLG\_ESTADO=1;

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[PRD\_LISTAR\_TIPO\_USUARIO] Script Date: 29/11/2020 00:40:46 \*\*\*\*\*\*/

GO

CREATE PROCEDURE [dbo].[PRD\_LISTAR\_TIPO\_USUARIO]

AS

BEGIN

SELECT A.ID\_TIPO\_USUARIO

,A.DESCRIPCION

FROM T\_MAE\_TIPO\_USUARIO A

WHERE A.FLG\_ESTADO = '1'

END;

--EXEC PRD\_LISTAR\_AULA @IN\_ID\_GRUPO = 1

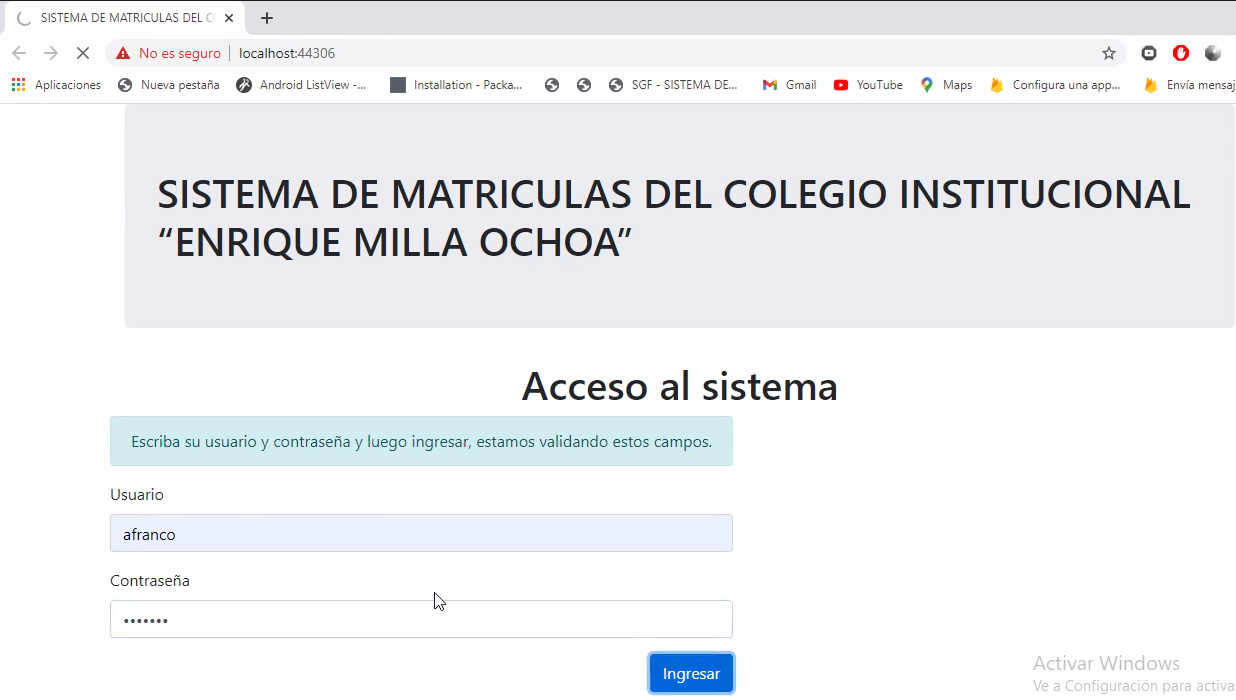
**Nota: Lista los tipos de usuario de la aplicación**

1. **DESARROLLO DE APLICACIÓN**
2. **Descripción de la Aplicación:**

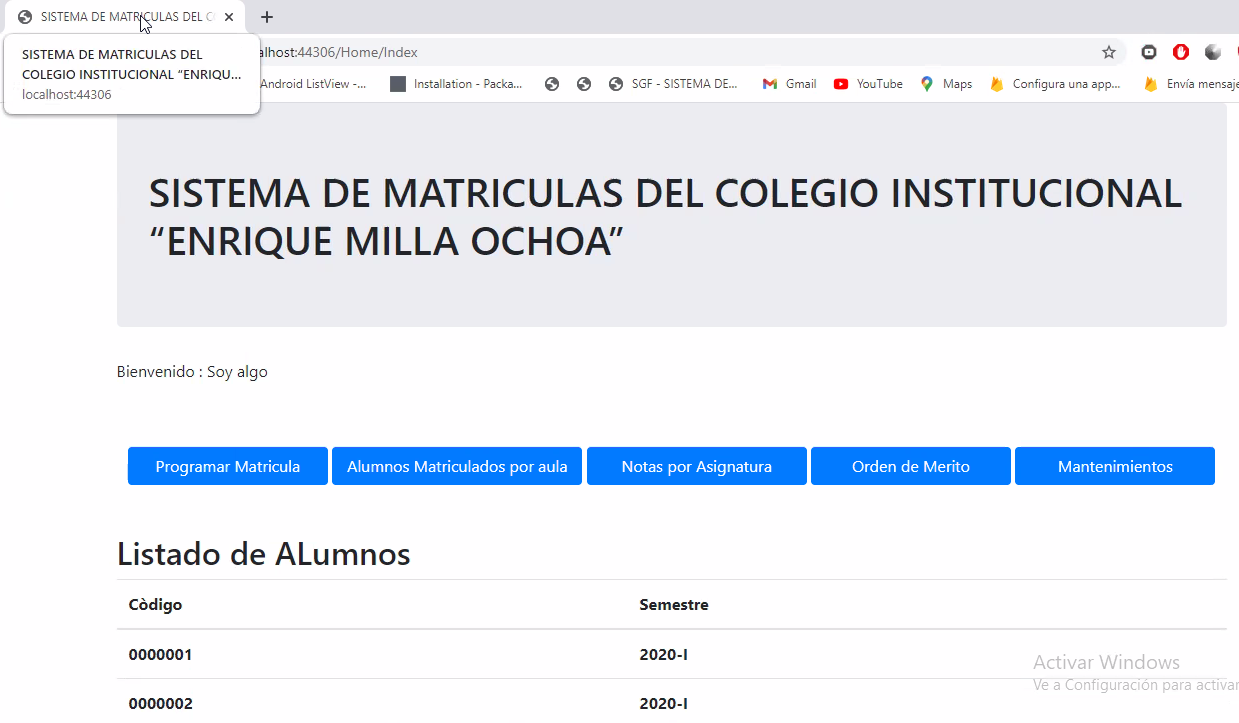
Para el desarrollo del sistema de matrícula del colegio Enrique Milla Ochoa; Se utilizo CSS, el Bootstrap Versión 4.5.3, el JQuery y Librerías propias creadas, También se usó Code First y Entity framework (Es un método para el manejo de la base de datos mediante entidades), MVC5- Modelo vista Controlador

1. **Diseño de Formularios (VISTAS)**

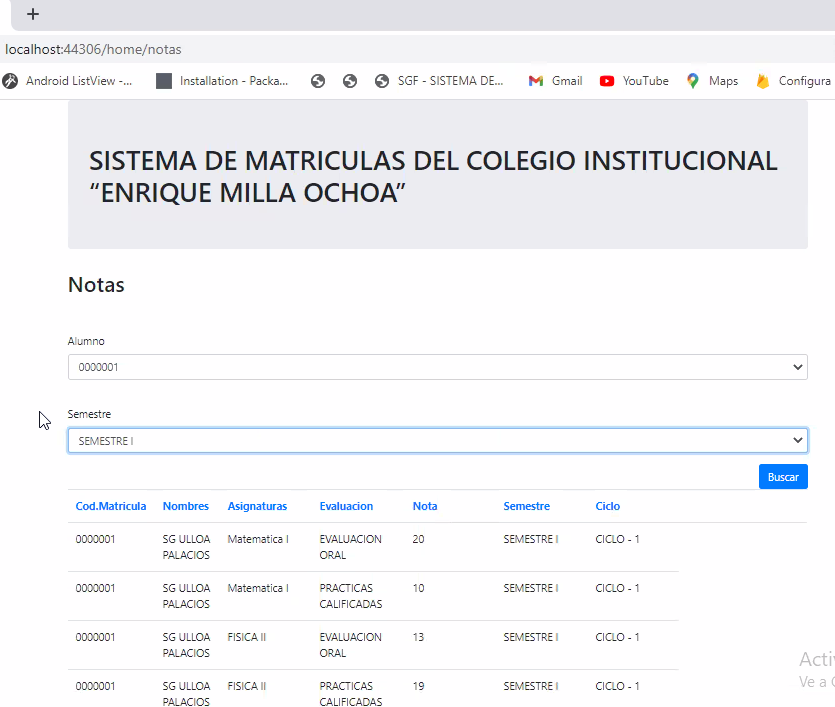
Para el desarrollo de los formularios del sistema de matrícula del colegio Enrique Milla Ochoa; Se utilizo los Helpers que ayudan a simplificar el código y organiza el llamado a controladores.



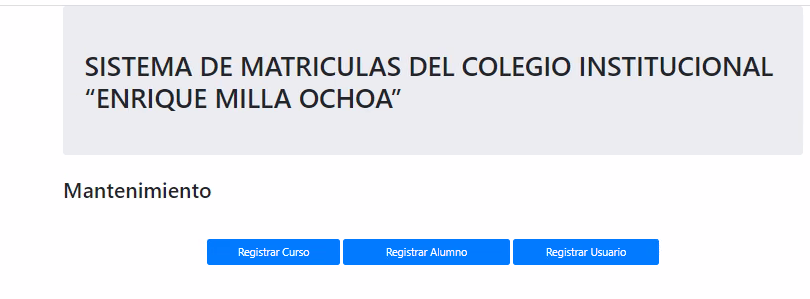
Formulario Inicio Sesión



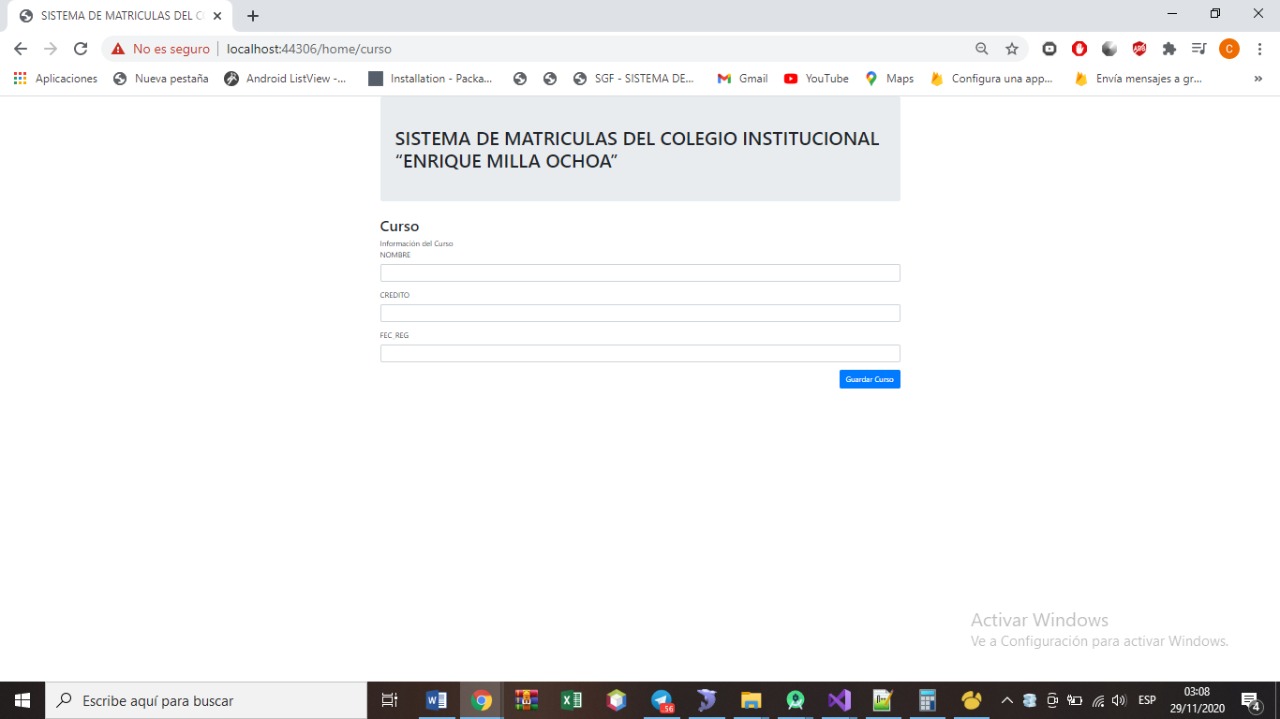
Formulario de Bienvenida



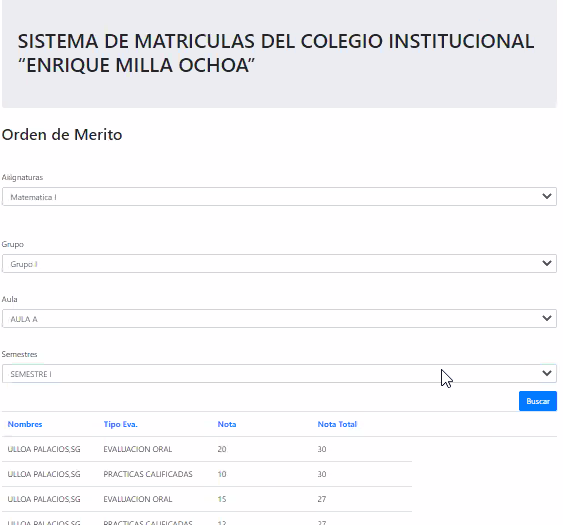
Formulario de Notas



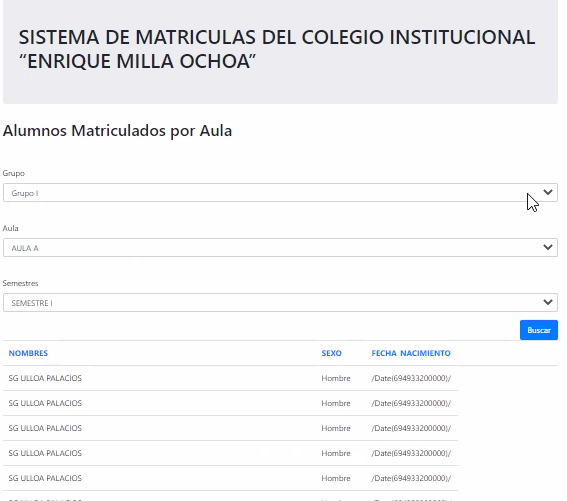
Mantenimientos



Mantenimientos de Curso

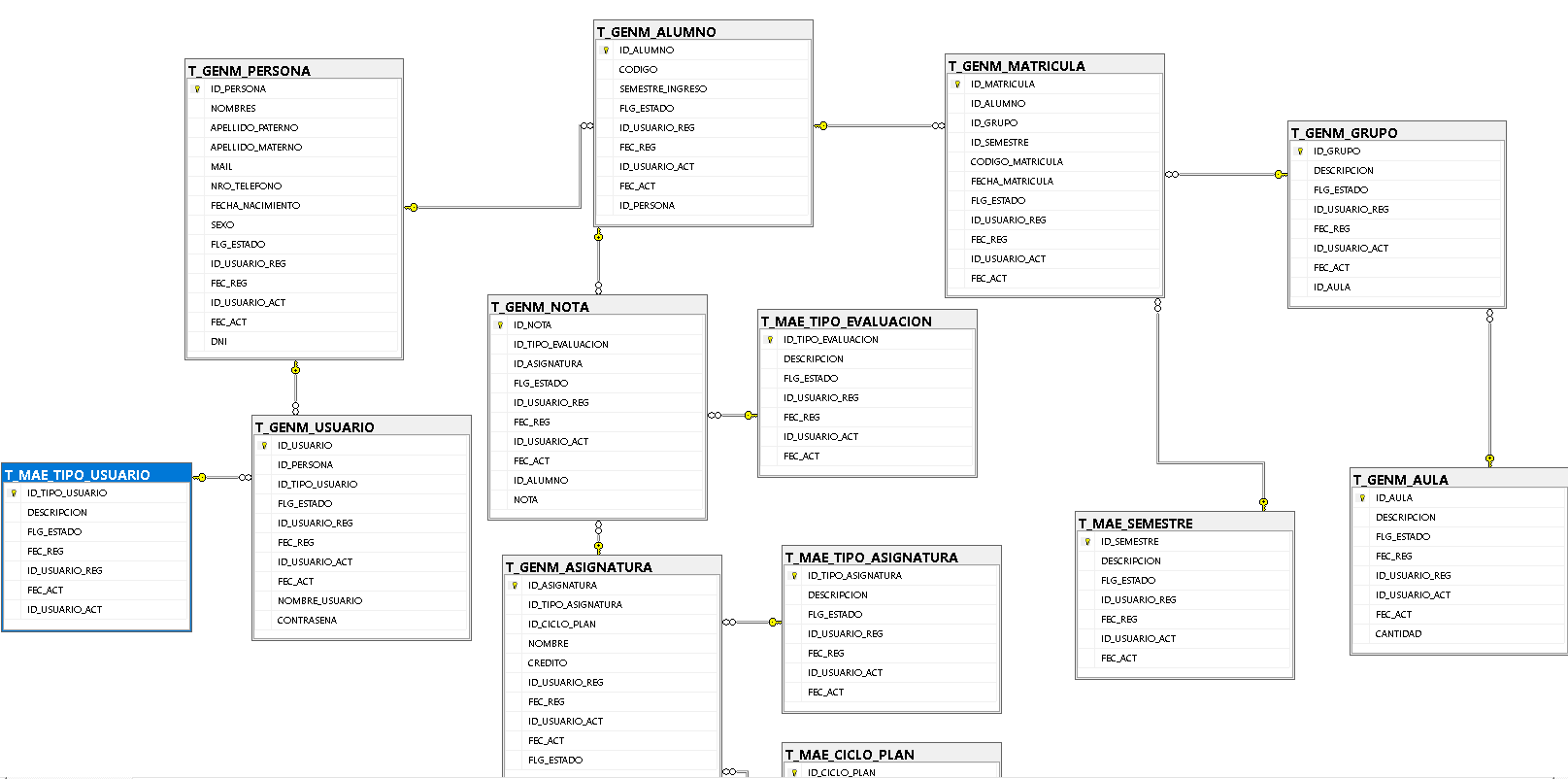


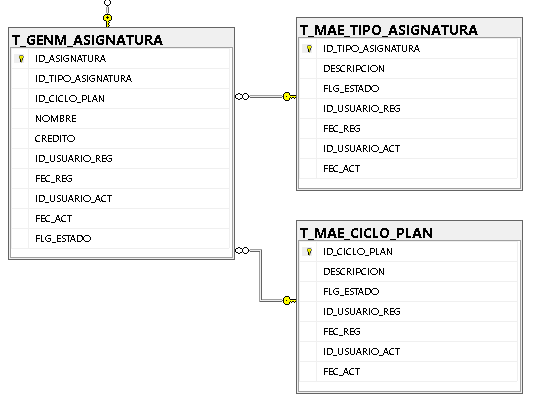
Formulario Orden de Merito



Formulario Alumnos Matriculado

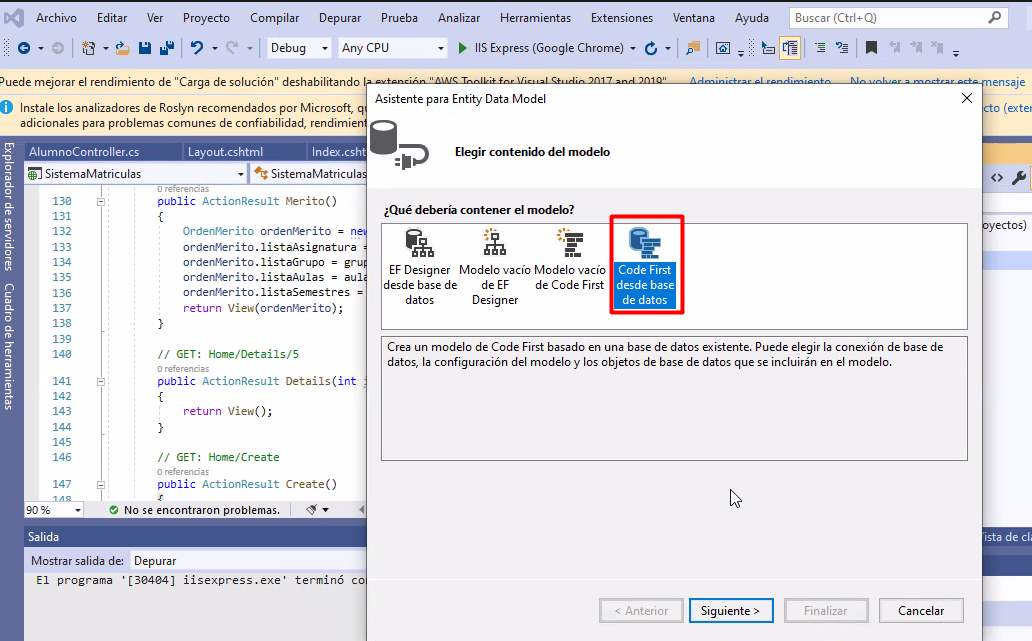
1. **DISEÑO DE BASE DATOS**

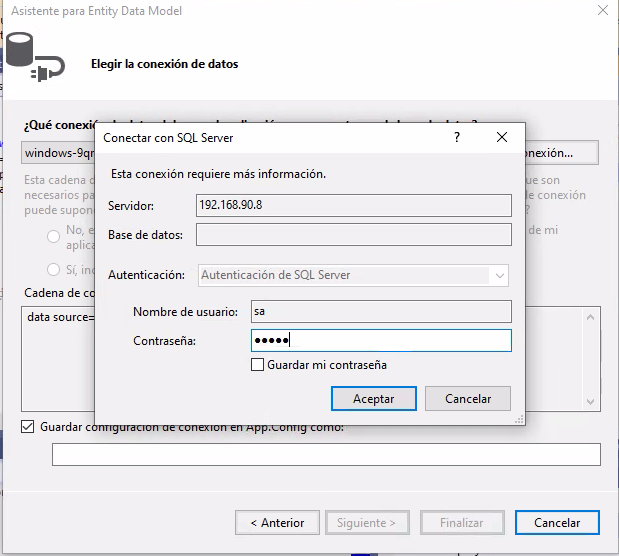


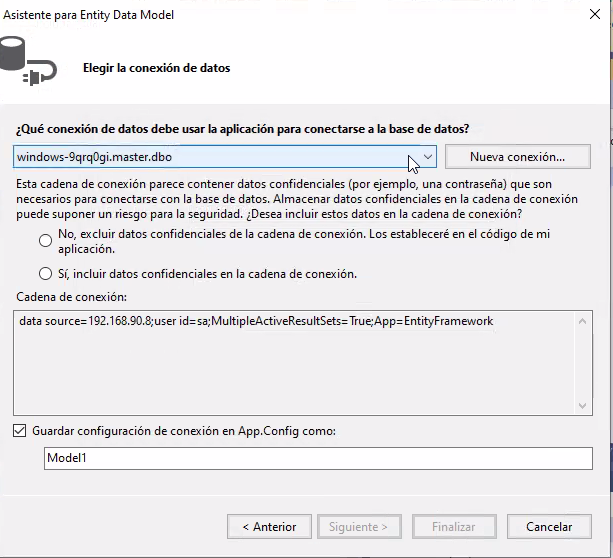


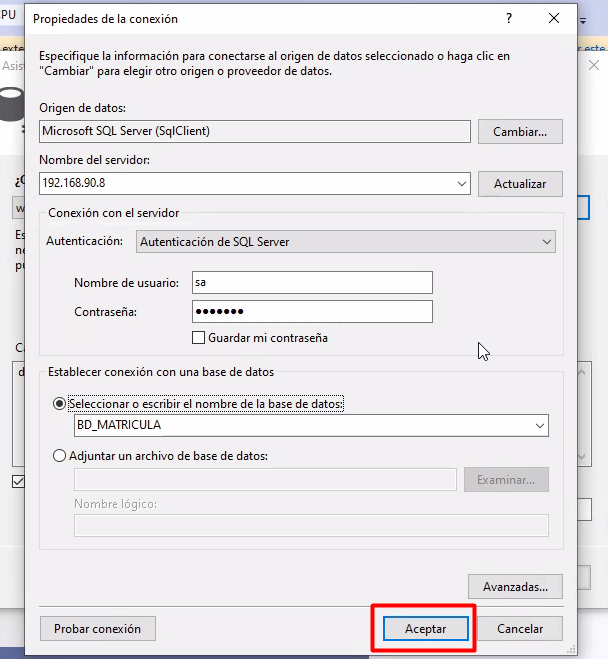
1. **Implementación de Code First y Entity framework**

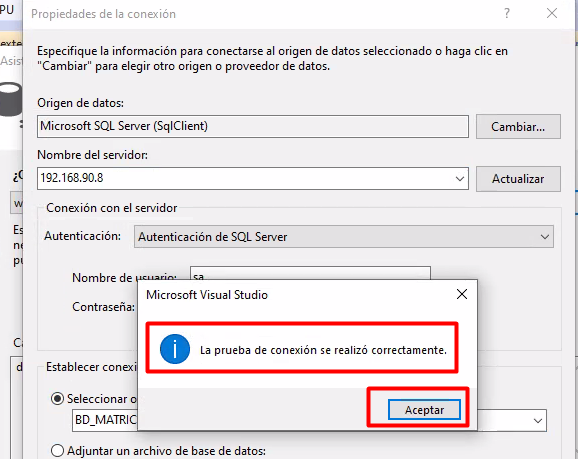
**Code First**

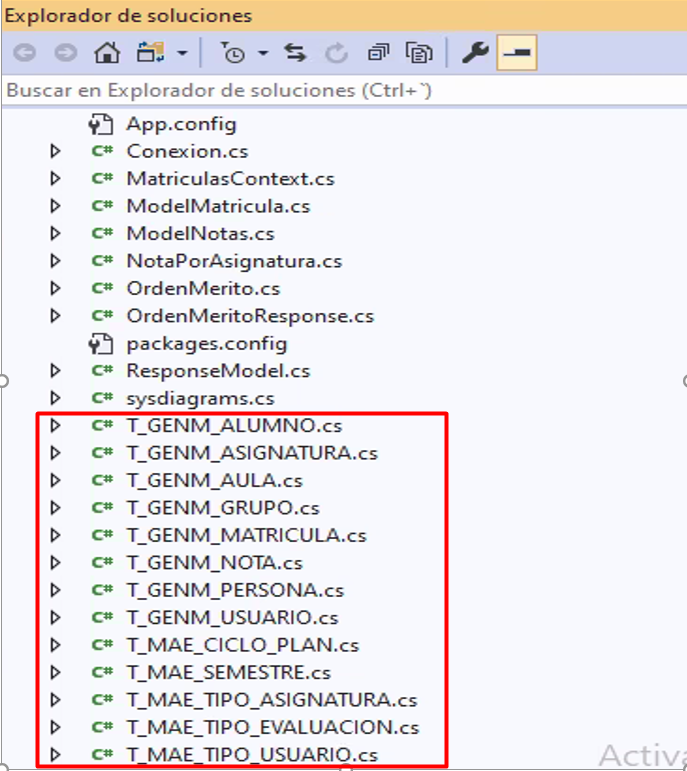












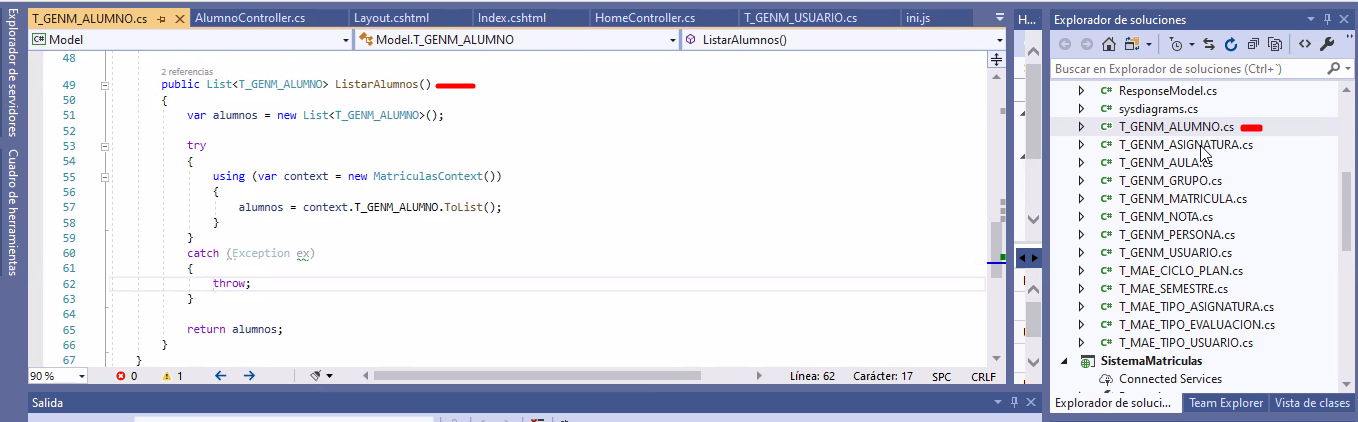
Implementación de Code First en la capa modelo del sistema de matricula

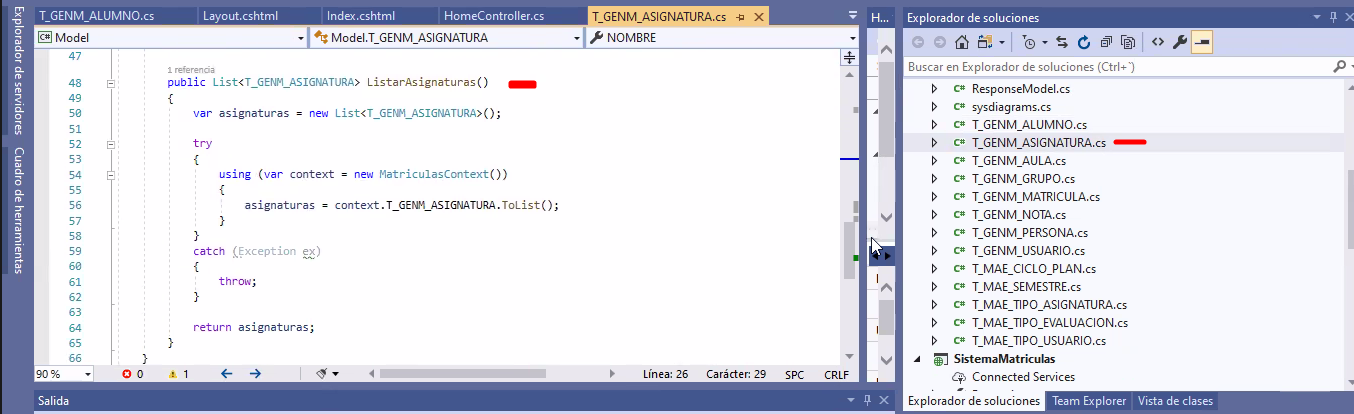
**Entity framework**

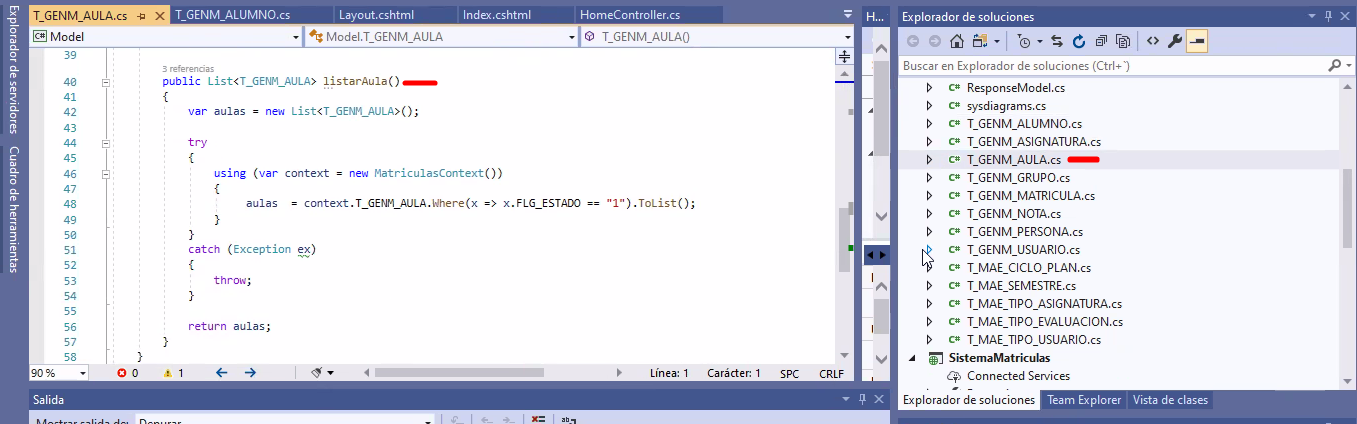
se puede trabajar consultas a la base de datos a través de métodos lambda;

Una expresión lambda es una expresión que devuelve un método

Un método típico está formado por cuatro elementos: El tipo que devuelve el método, • El nombre del método, • La lista de parámetros, y • El cuerpo del método donde están las instrucciones que se ejecutan







# **CONCLUSIONES**

El resultado del trabajo ha sido realmente muy provechoso ya que ha permitido tener una primera aproximación a la tecnología .NET, adquiriendo suficientes conocimientos como para desarrollar una aplicación Web de complejidad media.

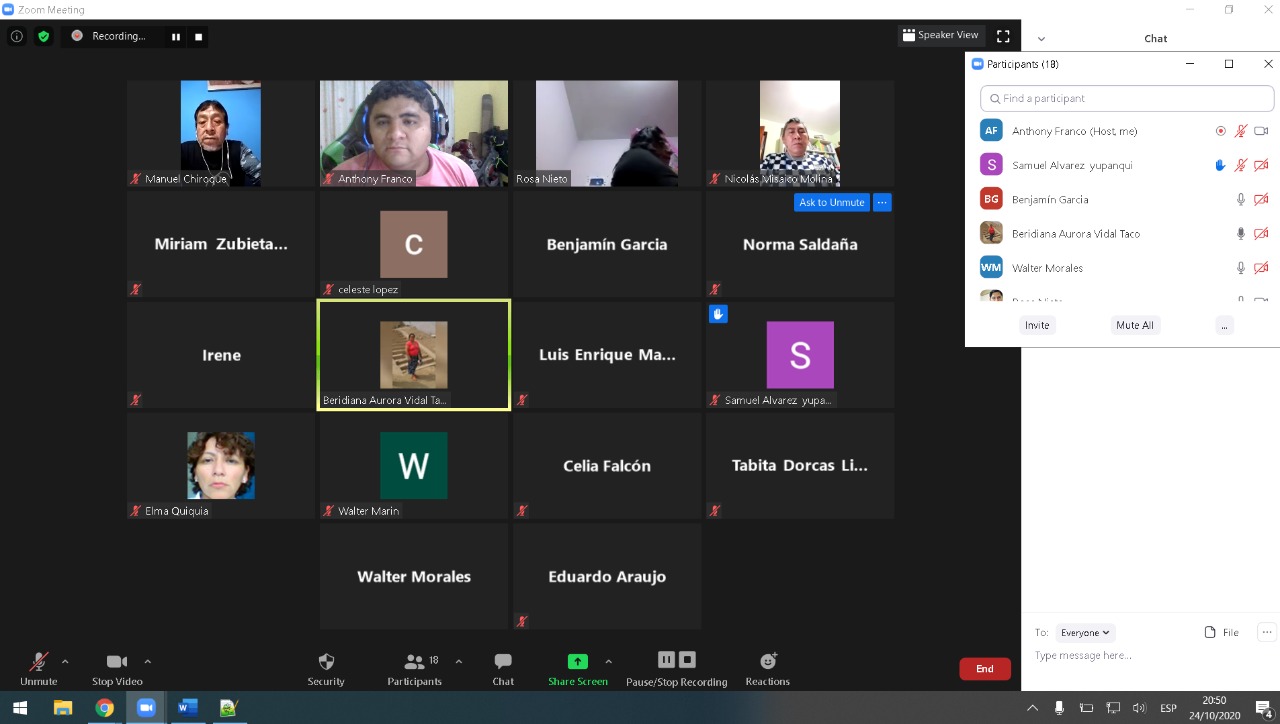
Además, ha servido como repaso y puesta en práctica de numerosos conceptos alcanzados en diversas asignaturas a lo largo de la carrera, especialmente de las asignaturas de programación, bases de datos y ingeniería del software. Se puede decir que se han alcanzado todos los objetivos propuestos inicialmente, tanto por el desarrollo de la aplicación como por los conocimientos adquiridos. Por otro lado, la propia redacción de la memoria es un ejercicio recomendable para alcanzar una base sólida en redacción de documentación técnica.

Podemos finalizar indicando también la importancia del desarrollo de aplicaciones en .NET, dado que este es multiplataforma y permite la ejecución del programa en diferentes entornos de producción asimismo se ha implementado métodos propios para dar mayor seguridad a la aplicación y de esta forma hacerlo más escalable por ejemplo la creación de Helpers para manejos de sesiones y la creación de una librería para el llenado de datos en una lista personalizada.

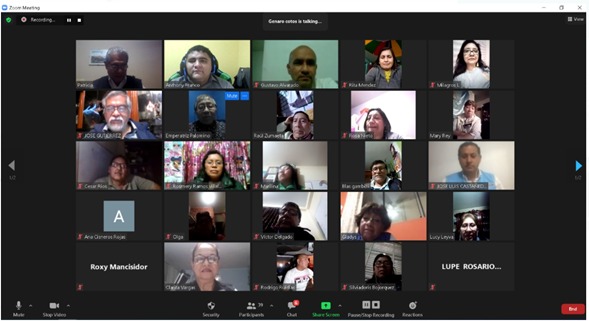
# **REFERENCIAS**

* Álvarez , M. (2020). Qué es la programación orientada a objetos . Obtenido de https://desarrolloweb.com/articulos/499.php
* Amoedo, D. (2015). Ubunlog. Obtenido de Ubunlog: https://ubunlog.com/mysql-workbench-bases-datos/
* EcuRed. (2020). Microsoft Visual Studio . Obtenido de https://www.ecured.cu/Microsoft\_Visual\_Studio
* Desarrolloweb. (02 de enero de 2014). https://desarrolloweb.com. Obtenido de https://desarrolloweb.com: https://desarrolloweb.com/articulos/que-es-mvc.html
* jdk oracle. (2015). https://www.java.com. Obtenido de https://www.java.com: https://www.java.com/es/download/faq/develop.xml
* Mestras, J. P. (Noviembre de 2008). https://www.fdi.ucm.es. Obtenido de https://www.fdi.ucm.es: https://www.fdi.ucm.es/profesor/jpavon/poo/2.14.MVC.pdf
* Microsoft. (2019). https://docs.microsoft.com. Obtenido de https://docs.microsoft.com: https://docs.microsoft.com/es-es/sql/relational-databases/stored-procedures/stored-procedures-database-engine?view=sql-server-ver15
* microsoft. (04 de 2019). https://www.microsoft.com/en-us/sql-server/sql-server-2017.
* uniwebsidad. (2016). https://uniwebsidad.com. Obtenido de https://uniwebsidad.com: https://uniwebsidad.com/libros/symfony-1-4/capitulo-2/el-patron-mvc
* Expresiones lambda (Guía de programación de C#) http://msdn.microsoft.com/es-es/library/bb397687.aspx
* Árboles de expresión (C# y Visual Basic) http://msdn.microsoft.com/es-es/library/bb397951.aspx
* Las expresiones lambda en C# 3.0 http://www.elguille.info/NET/futuro/firmas\_octavio\_ExpresionesLambda.htm
* C# es fácil (iv): Expresiones lambda http://burbujasnet.blogspot.com.es/2008/06/c-es-fcil-iv-expresiones-lambda.html
* Expresiones lambda <http://geeks.ms/blogs/gtorres/archive/2010/02/22/expresiones-lambda.aspx>

# **ANEXO**



Reunión Virtual Recopilando información del Colegio

****

Reunión Virtual con el Colegio, Personal Administrativo y Representante de UGEL