

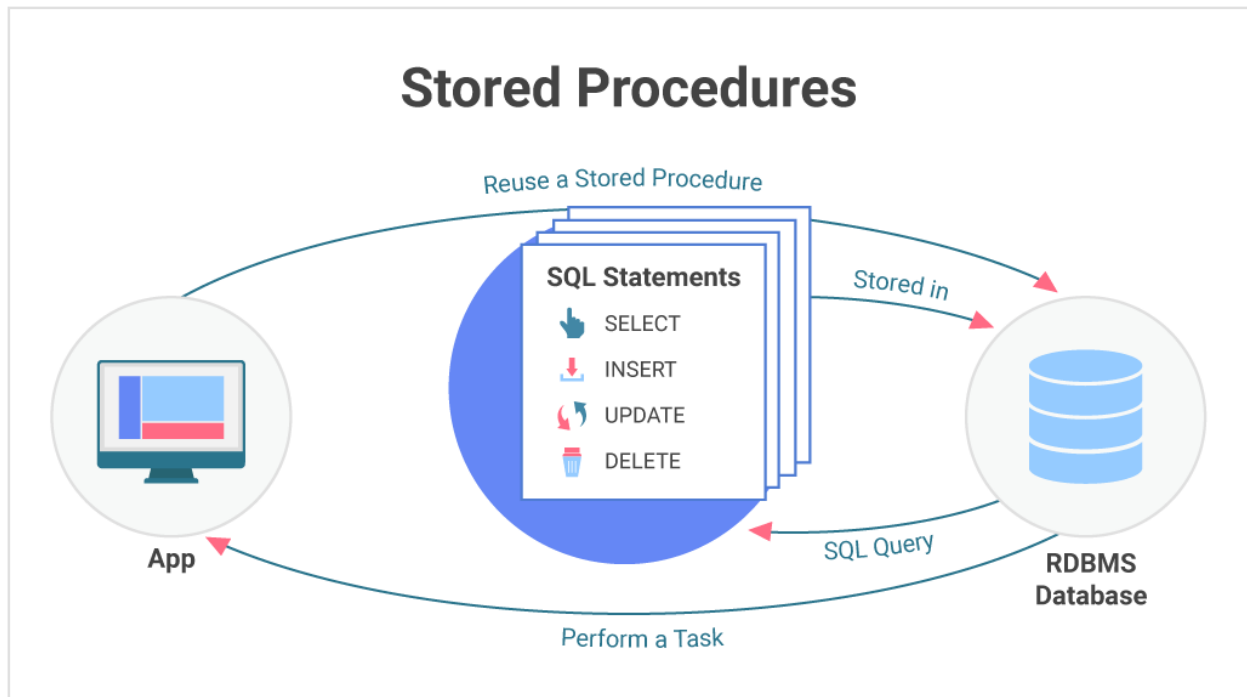
TRABAJO ACADEMICO Nro. 12

Escuela Profesional:	Ingeniería de Sistemas		Asignatura:	Gestión de Datos e Información II	
Ciclo:	V	Turno:	Mañana/Tarde	Semestre Académico:	2022-1
Docentes:	Ing. Eric Gustavo Coronel Castillo		Fecha:	Del 20/06/2022 al 25/06/2022	

Ejecución de Procedimientos Almacenados desde Visual Studio

I. INTRODUCCION	2
II. OBJETIVO	3
III. METODOLOGIA Y ACTIVIDADES	3
IV. RECURSOS.....	3
V. OBSERVACIONES.....	4
VI. REQUERIMIENTO A RESOLVER	4
Modelo educativo: EDUCA	4
VII. REQUERIMIENTO A RESOLVER	6
Requerimiento 1	6
Requerimiento 2	6
Requerimiento 3	7

I. INTRODUCCION



Los procedimientos almacenados son muy utilizados para implementar lógica de negocio en la base de datos.

En esta oportunidad implementaras lógica de negocio en procedimientos almacenados y lo ejecutaras desde Visual Studio utilizando el lenguaje C#.

II. OBJETIVO

Elaborar lógica de negocio en procedimientos almacenados y ejecutarlos desde una aplicación de escritorio desarrollada con Visual Studio y C#.

III. METODOLOGIA Y ACTIVIDADES

Debe seguir los siguientes pasos para cada requerimiento:

1. Analizar el requerimiento.
2. Analizar el procedimiento utilizando la técnica de la caja negra.
3. Implementar el procedimiento almacenado.
4. Implementar la ejecución del procedimiento almacenado desde la capa de servicios de la solución con Visual Studio.
5. Implementar la interface de usuario en la capa view de la solución con Visual Studio.
6. Integrar la capa view con la capa de servicios.
7. Verifique que funciona correctamente.
8. Recoger evidencias.
9. Elaborar informe que incluya conclusiones y recomendaciones.

IV. RECURSOS

1. Acceso a internet.
2. Computadora personal con Windows 10 o superior
3. Microsoft SQL Server correctamente instalado.
4. Microsoft SQL Server Management Studio
5. Visual Studio
6. Plataforma Blackboard para la entrega de su solución.

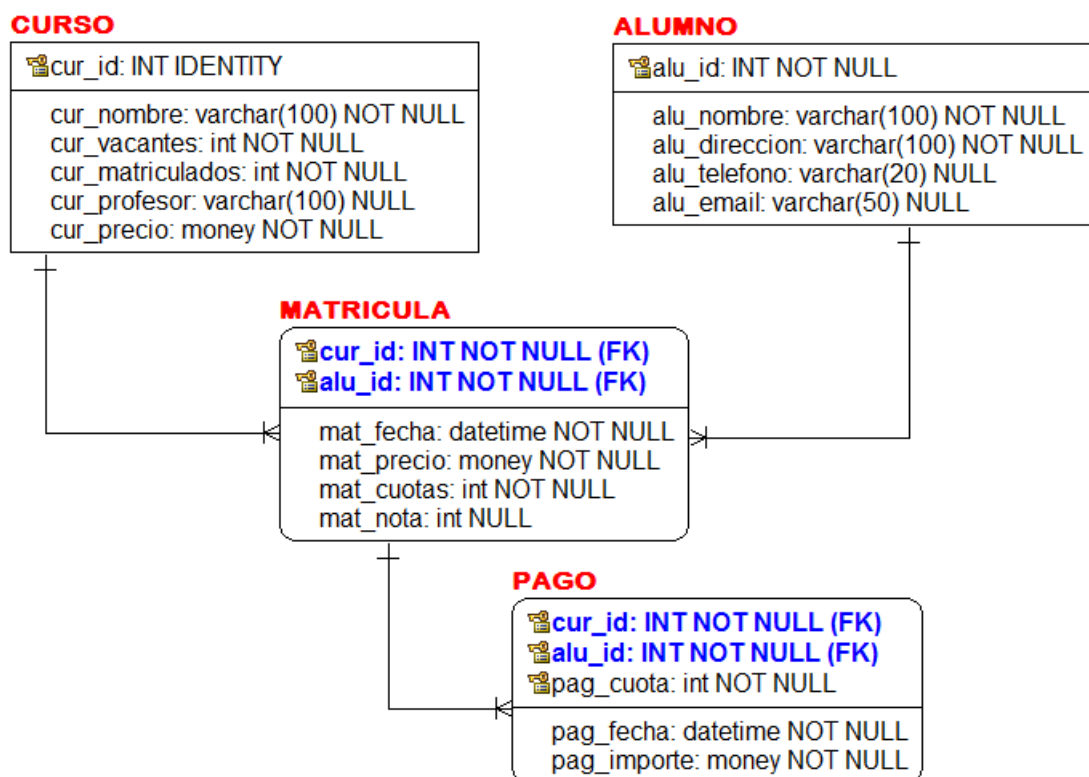
V. OBSERVACIONES

1. Se debe crear una **carpeta de trabajo** con el nombre "**GDI2\GPO##TA11**", en donde grabara todos los archivos de este trabajo académico. Donde **##** representa el número de grupo, por **01**, **02**, **03**, y así sucesivamente.
2. Para finalizar, debe empaquetar la carpeta **GPO##TA09** en un archivo **RAR** o **ZIP**, este archivo empaquetado es el que debe subir a la plataforma Blackboard en la sección correspondiente.

VI. REQUERIMIENTO A RESOLVER

Modelo educativo: EDUCA

Este modelo de base de datos corresponde a un modelo básico de un sistema de matrícula.



A continuación, tienes una breve descripción de las tablas.

- En la tabla **CURSO** se tienen los cursos programados, en esta tabla se tiene el precio de lista, las vacantes programadas y la cantidad de matriculados. Cada vez que se matricula un nuevo alumno, la columna matriculados debe incrementarse en 1.

- En la tabla **ALUMNO** se registran los alumnos, es muy importante registrarlos de lo contrario no se pueden matricular, quiere decir que es necesarios registrarlo antes de que intenten matricularse en algún curso.
- En la tabla **MATRICULA** se registran las matrículas, los datos muy importantes aquí son el precio y la cantidad de cuotas, el precio no puede ser mayor al precio de lista y tampoco menor al 50% del precio de lista, el precio de lista lo tienes en la tabla **CURSO**, la cantidad de cuotas puede ser:

CUOTAS	DESCRIPCIÓN
1	En este caso se trata de pago al contado, al momento de la matricula se debe registrar también el pago en la tabla PAGO . Es posible que el precio que pagará el alumno sea menor al precio de lista, pero no menos del 50%.
2	En este caso cada cuota es el 50% del precio, al momento de la matricula se debe registrar el pago de la primera cuota. En este caso el precio que pagará el alumno debe ser igual al precio de lista.
3	En este caso, el precio que pagará el alumno debe ser igual al precio de lista. La primera cuota será el 40% del precio, y las otras 2 cuotas serán el 30% del precio cada una. Durante la matricula se debe registrar el pago de la primera cuota.

- En la tabla **PAGO** se registran los pagos, si un curso se paga al contado (1 cuota) en esta tabla de registrarse un solo pago. Si el curso se paga en 2 cuotas, esta tabla debe registrarse 2 pagos en momentos diferentes, la suma de las 2 cuotas debe ser igual al precio del curso. Si el curso se paga en 3 cuotas, esta tabla debe registrarse 3 pagos en momentos diferentes, la suma de las 3 cuotas debe ser igual al precio del curso.

El script para crear esta base de datos lo puedes descargar desde la siguiente URL:

<https://github.com/gcoronelc/databases>

VII. REQUERIMIENTO A RESOLVER

Para desarrollar estos requerimientos debe considerar la descripción de la base de datos realizada en el punto anterior.

Se debe implementar una sola solución con Visual Studio para todos los requerimientos.

Requerimiento 1

Registro de nuevos alumnos (6 Puntos)

1. Desarrollar un procedimiento almacenado para registrar un nuevo alumno. El procedimiento de retornar el ID del alumno.
2. Implementar el procedimiento almacenado en la capa de servicios de la solución desarrollada con Visual Studio.
3. Implementar la interfaz de usuario en la capa de presentación o view de la solución desarrollada con Visual Studio.
4. Integrar la capa de presentación y la capa de servicios.

Requerimiento 2 (8 Puntos)

Registrar una nueva matricula

1. Desarrollar un procedimiento almacenado para registrar una nueva matricula.
2. Implementar el procedimiento almacenado en la capa de servicios de la solución desarrollada con Visual Studio.
3. Implementar la interfaz de usuario en la capa de presentación o view de la solución desarrollada con Visual Studio.
4. Integrar la capa de presentación y la capa de servicios.

Requerimiento 3 (6 Puntos)

Consulta de alumnos matriculados

1. Desarrollar un procedimiento almacenado para consultar los alumnos matriculados en un curso específico.
2. Implementar el procedimiento almacenado en la capa de servicios de la solución desarrollada con Visual Studio.
3. Implementar la interfaz de usuario en la capa de presentación o view de la solución desarrollada con Visual Studio.
4. Integrar la capa de presentación y la capa de servicios.