

# **Universidad Nacional**

## **Sede Regional Brunca**



**Programación II**

**Proyecto #2**

**NoteBook**

**Estudiantes:**

**Jeffry Zúñiga Obando**

**Luis Miguel Valverde Navarro**

**Profesor:**

**Rubén Mora Vargas**

**III Ciclo, 2019**

## Tabla de contenido

### Contenido

Introducción .....	3
Planteamiento del problema .....	4
Solución adaptada .....	4
<b>ERS (Especificación de Requerimientos) .....</b>	<b>5</b>
Conclusión .....	9

## Introducción

En este presente proyecto se realizará por medio de visual Studio, utilizando C# como lenguaje de programación.

Este sistema que va contar con una restricción de acceso por medio de un logueo, donde sino existe ningún usuario, se podrá agregar uno desde el formulario de logueo, en el caso de que ya exista un usuario de tipo “Administrador” el podrá realizar la función de agregar nuevos usuarios desde el menú principal y estos serán guardados en una base de datos.

El usuario tendrá la opción de crear cuadernos con características tales como: nombre, categoría, color y orden, donde cada cuaderno ya guardado tendrá la opción de agregar notas con las siguientes propiedades: titulo, color, fuente, color de letra, orden, fecha de creación y fecha de modificación. Para una mayor facilidad de uso del sistema se agregó una función de búsqueda tanto para los cuadernos como para las notas. Además, el programa deberá ir realizando un registro de actividades del usuario que esta haciendo uso del sistema, el registro va guardar una fecha y hora, usuario, acción, objeto e información adicional.

### **Planteamiento del problema**

Se debe desarrollar un software que permita realizar la autenticación del usuario para ingresar al sistema o en caso de que no exista usuarios, se deberá de crear uno nuevo. Dicho sistema debe almacenar cuadernos con sus respectivas propiedades en una base de datos, así mismo los cuadernos podrán contener ninguna o muchas notas, donde cada nota tendrá sus respectivas propiedades que la identifiquen. Toda la información de dicho sistema se guardará de manera permanente en un base de datos, por lo que el usuario podrá ver sus cuadernos y notas cuando él desee. De una manera automática el sistema debe de generar un registro de actividades realizadas por el usuario que inicializo el sistema.

Las contraseñas de todos los usuarios y las notas que especifiquen su privacidad de tipo privada deberán ser encriptadas al momento de almacenarlas en la base de datos.

Se debe crear una base de datos que contenga todas las tablas necesarias para almacenar toda la información ingresada al sistema de NoteBook, así de esta manera la información ingresada en un momento dado podrá ser recupera en cualquier momento que se esté utilizando el sistema.

### **Solución adaptada**

Se creo una base de datos con todas las tablas necesarias, así como sus propiedades, para que pueda almacenar toda la información generada por el sistema. Realizamos la conexión de visual studio con MySQL por medio de dos clases llamadas: "MysqlAccess" y "DbAccess", donde por medio de ellas y sus métodos enviaremos y buscaremos información a la base de datos.

Se creo la opción de ingresar un usuario al inicio del programa ya que al primer momento del uso de este sistema no va a contar con ningún usuario guardado en la base de datos, seguidamente realizaremos la autenticación del usuario ingresado al sistema con la base de datos, donde si efectivamente hay una coincidencia se le da el acceso y podrá ser uso de todo el sistema, en caso de que el usuario sea usuario de tipo "Administrador", en caso de que sea tipo "Regular", tendrá algunas restricciones.

Las contraseñas y las notas privadas serán encriptadas y desenscriptadas por medio de una clase llamada "Enpriptar".

Se podrá ingresar todos los cuadernos que se deseen, de modo que cada cuaderno pueda contener ninguna, una o muchas notas, donde todas sus propiedades van a ser guardadas en la base de datos y mostradas en un dataGridView con sus propiedades. También se podrán modificar y eliminar la información de los cuadernos y las notas. También se podrá realizar la búsqueda aproximada por medio del nombre del cuaderno o por el nombre de la nota.

## ERS (Especificación de Requerimientos)

### Requerimiento persistencia

Persistencia de visual studio con MySQL

<b>Requerimiento</b>	REQ08	<b>Estado</b>	Aceptable
<b>Solicitante</b>	Luis Valverde.	<b>Puesto</b>	Estudiante
<b>Descripción</b>	<b>Módulo</b>	Persistencia a la base de datos	
El sistema debe presentar una persistencia con una base de datos para que sus datos sean almacenados de forma permanente.			
<b>Especificación</b>			
Este requerimiento se llevará a cabo por medio de dos clases llamadas “MysqlAccess” y “DbAccess”, donde contiene los métodos para enviarle y consultar información a la base de datos de MySQL.			
<b>Justificación</b>			
Este requerimiento se justifica en su elaboración, para realizar el almacenamiento de una forma permanente.			

### Requerimiento encriptación

Encriptación de contraseñas y notas privadas.

Requerimiento	REQ09	Estado	Aceptable
Solicitante	Luis Valverde.	Puesto	Estudiante
Descripción	Módulo	Encriptación.	
El sistema deberá encriptar las contraseñas y las notas privadas que serán guardadas en la base de datos y sean accesibles solo por el usuario creador.			
Especificación			
Este requerimiento se llevará a cabo por medio de una clase llamada “encriptar”, donde tendrá los métodos de encriptar y desencriptar, y recibirán un string para realizar su función.			
Justificación			
Este requerimiento se justifica en su elaboración, para proteger las contraseñas y sus notas de usuarios que puedan acceder a la base de datos.			

### Requerimiento Logueo.

Autenticación en el sistema.

Requerimiento	REQ10.1	Estado	Aceptable
Solicitante	Jeffry Zúñiga.	Puesto	Estudiante
Descripción	Módulo	Autenticarse en el sistema	
El sistema al tener un acceso restringido, deberá de verificar si el usuario ingresado se encuentra en la base de datos, y así podrá acceder al sistema.			
Especificación			
Este requerimiento se llevará a cabo por medio de una tabla en la base de datos llamada usuarios, y el sistema consultará dicha tabla con los valores que el usuario ingreso en el sistema.			
Justificación			
Este requerimiento se justifica en su elaboración, para mantener la información de cada usuario separada y que solamente se posible su modificación a través del usuario creador.			

### Requerimiento cambio de contraseña.

Un usuario podrá realizar el cambio de su contraseña.

Requerimiento	REQ10.2	Estado	Aceptable
Solicitante	Jeffry Zúñiga	Puesto	Estudiante
Descripción	Módulo	Cambiar contraseña	
El sistema deberá ser capaz de modificar la contraseña existente en la base de datos, sin afectar la información del usuario.			
Especificación			
Este requerimiento se llevará a cabo por medio de la conexión a la base de datos, donde obtendremos la tabla de usuarios y dentro de ahí buscaremos la coincidencia del usuario actual con todos los usuarios ingresados, realizaremos el cambio.			
Justificación			
Este requerimiento se justifica en su elaboración, ya que algunas veces los usuarios por cuestión de seguridad intentan mejorar la contraseña o ingresan una contraseña más fácil de recordar.			

### Requerimiento agregar, modificar y eliminar cuadernos.

El usuario podrá agregar, modificar y eliminar cuadernos de su creación.

Requerimiento	REQ10.3	Estado	Aceptable
Solicitante	Jeffry Zúñiga.	Puesto	Estudiante
Descripción	Módulo	Agregar, modificar y eliminar cuadernos	
El sistema deberá agregar, modificar y eliminar cuadernos digitales del usuario registrado.			
Especificación			
Este requerimiento se llevará a cabo por medio de la conexión con la base de datos, donde obtenemos los idCuadernos para luego así ya identificado el cuaderno eliminarlo o editarlo, en caso de agregar es nada más de insertar un nuevo cuaderno.			
Justificación			
Este requerimiento se justifica en su elaboración, ya que el usuario pueda tener necesidades de guardar muchos cuadernos con diferente contenido, o desea eliminar un cuaderno que ya no tiene uso e incluso modificar información ya existente.			

### Requerimiento agregar, modificar y eliminar notas.

El usuario podrá agregar, modificar y eliminar notas de su creación.

Requerimiento	REQ10.4	Estado	Aceptable
Solicitante	Jeffry Zúñiga.	Puesto	Estudiante
Descripción	Módulo	Agregar, modificar y eliminar notas	
El sistema deberá agregar, modificar y eliminar notas digitales del usuario registrado.			
Especificación			
Este requerimiento se llevará a cabo por medio de la conexión con la base de datos, donde obtenemos los idNota para luego así ya identificado la nota eliminarlo o editarlo, en caso de agregar es nada más de insertar una nueva nota.			
Justificación			
Este requerimiento se justifica en su elaboración, ya que el usuario pueda tener necesidades de guardar muchas notas con diferente contenido, o desea eliminar una nota que ya no tiene utilidad e incluso modificar información ya existente.			

### Requerimiento historial de acciones.

Realiza un historial con todas las acciones realizadas por el usuario.

Requerimiento	REQ10.5	Estado	Aceptable
Solicitante	Luis Valverde.	Puesto	Estudiante
Descripción	Módulo	Historial de acciones	
El sistema deberá registrar las acciones que se realicen en el, por cada usuario para poder ver un historial de su actividad en la aplicación, se debe de guardar: fecha y hora, usuario, acción, objetivo y información adicional.			
Especificación			
Este requerimiento se llevará a cabo por medio de la conexión con la base de datos donde existe una tabla que guardara todas las acciones realizadas por el usuario, así mismo serán mostradas en un datagridview.			
Justificación			
Este requerimiento se justifica en su elaboración, ya que se necesita poder almacenar los datos de una manera permanente por si sucede un inconveniente tener un registro de todas las acciones que se realizaron.			

### Requerimiento conexión a base de datos.

Conexión de visual studio con MySQL con una clase.

<b>Requerimiento</b>	REQ10.6	<b>Estado</b>	Aceptable
<b>Solicitante</b>	Luis Valverde.	<b>Puesto</b>	Estudiante
<b>Descripción</b>	<b>Módulo</b>	Conexión a la base de datos.	
El sistema tendrá una conexión a una base de datos, donde su información debe estar persistida de manera permanente.			
<b>Especificación</b>			
Este requerimiento se llevó a cabo por medio de dos clases, donde por medio de sus métodos se puede insertar y obtener información de la base de datos.			
<b>Justificación</b>			
Este requerimiento se justifica en su elaboración, para tener la información de una manera permanente en el sistema.			

### Requerimiento búsqueda.

Se realiza la búsqueda por medio del nombre del cuaderno y por el título de la nota.

Requerimiento	REQ10.7	Estado	Aceptable
Solicitante	Luis Valverde.	Puesto	Estudiante
Descripción	Módulo	Búsqueda	
El sistema deberá tener la opción de búsqueda para notas de texto en el usuario que se haya registrado, la búsqueda se realizará por medio de una palabra relaciona, también se realizará la búsqueda por medio de cuadernos, y se debe mostrar el resultado de cada búsqueda.			
Especificación			
Este requerimiento se llevará a cabo por medio de la conexión con la base de datos donde se busca primeramente solo los cuadernos o notas con el id del usuario autenticado y después de ello se mostrará en un datagridview los cuadernos o notas filtradas.			
Justificación			
Este requerimiento se justifica en su elaboración, ya que, a la hora de tener muchos cuadernos, el usuario se le dificulta la búsqueda de información, por lo cual este requerimiento es de gran ayuda a la comunicación usuario-sistema.			

### Requerimiento acceso gradual.

Algunos usuarios tienen funcionalidades limitadas.

Requerimiento	REQ11	Estado	Aceptable
Solicitante	Jeffry Zúñiga.	Puesto	Estudiante
Descripción	Módulo	Acceso gradual.	
El sistema deberá manejar el acceso a las funciones de forma gradual, ósea debe tener permiso para cada acción que realice en el sistema.			
Especificación			
Este requerimiento se llevará a cabo por medio de los permisos que serán otorgado al momento de la creación del usuario, ya que se deberá especificar el tipo de usuario, administrador o regular, donde el primero tiene un acceso total al sistema mientras que el segundo tiene unas pequeñas restricciones			
Justificación			
Este requerimiento se justifica en su elaboración, ya que algunos usuarios no de menor rango y no necesitan realizar dichas funciones por lo cual de este modo se limitan ciertos usuarios a realizar acciones innecesarias.			

### Requerimiento crear usuarios.

Se podrán crear nuevos usuarios con su propia colección de cuadernos y notas.

<b>Requerimiento</b>	REQ12	<b>Estado</b>	Aceptable
<b>Solicitante</b>	Jeffry Zúñiga.	<b>Puesto</b>	Estudiante
<b>Descripción</b>	<b>Módulo</b>	Crear usuarios	
El sistema deberá tener un apartado de mantenimiento de usuarios donde se podrán crear nuevos usuarios o cambiar de contraseña.			
<b>Especificación</b>			
Este requerimiento se llevará a cabo por medio del almacenamiento de un nombre de usuario, nombre real, contraseña y tipo de usuario, donde dichas propiedades serán guardadas en la base de datos.			
<b>Justificación</b>			
Este requerimiento se justifica en su elaboración, para que cada usuario pueda tener su propia colección de cuadernos y notas.			

### Requisito refactorización.

Mejora de código.

Requerimiento	REQ13	Estado	Aceptable
Solicitante	Jeffry Zúñiga.	Puesto	Estudiante
Descripción	Módulo	Refactorización.	
El sistema deberá contar con una pequeña explicación de algunas técnicas de refactorización utilizaron.			
Especificación			
Este requerimiento se llevará a cabo explicando el código que estaba antes, que técnica de refactorización utilizaron y porque, así como el código nuevo.			
Justificación			
Este requerimiento se justifica en su elaboración, para dar a conocer algunas mejoras que se le pueden realizar al sistema.			



## **Conclusión**

Al finalizar este proyecto podemos concluir lo siguiente:

- La información de este sistema debía ser de forma permanente, de manera que se realizo la persistencia con una base de datos, (MySQL) donde tuvimos que investigar sus funcionamientos y propiedades, para así brindarle un mejor funcionamiento al sistema.
- Al ser un proyecto con funcionalidades meramente enfocadas a un entorno laboral, nos queda un gran aprendizaje, en la importancia de la seguridad por medio de un usuario y además de ello, tener un historial de actividades es sumamente esencial a la hora de hallar un inconveniente.
- El sistema fue creado por medio del lenguaje de programación C#, donde obtuvimos un gran conocimiento acerca de este lenguaje, ya que como futuros programadores debemos de tener al menos un poco de experiencia en algunos de los lenguajes de programación más reconocidos.