****

**UNIVERSIDAD TECNOLÓGICA DE EL SALVADOR FACULTAD DE INFORMÁTICA**

**Y CIENCIAS APLICADAS**

**ESCUELA DE INFORMÁTICA**

**PROGRAMACIÓN III, SECCIÓN 01**

**DOCENTE:**

JORGE ALBERTO ACEVEDO DÍAZ

|  |  |
| --- | --- |
| **ESTUDIANTES:** | **CARNET:** |
| ALVARADO FLORES, DANIEL ALEJANDRO | 25-1664-2020 |
| ASCENCIO ZÚNIGA, MADELINE NICOLE | 17-3762-2020 |
| CUATRO RIVERA, FERNANDO MIGUEL | 25-0871-2020 |
| ECHEVERRÍA SANTACRUZ, DIEGO ARMANDO | 25-0061-2020 |
| HERNÁNDEZ ROLIN, LIZBETH JAZMIN | 17-0829-2020 |
| LOPEZ ANDRES, JOSE TRANSITO | 17-2444-2020 |
| MARTÍNEZ CORTEZ, CARLOS GUILLERMO | 17-1543-2020 |
| MENJIVAR MARROQUIN, ROGELIO ISAÍ | 25-2786-2020 |
| MONTANO MUÑOZ, ERIKA ROXANA | 27-2005-2020 |
| SIGUENZA ORELLANA, CHRISTIAN JEFFRY | 27-0034-2020 |
| VÁSQUEZ CASTELLANOS, CARLOS RODOLFO | 25-0828-2019 |
|  |  |

**FECHA DE ENTREGA: LUNES 30 DE MAYO DE 2022**

# **Índice**

[Introducción 4](#_Toc104783360)

[Problema Investigado 5](#_Toc104783361)

[Justificación 6](#_Toc104783362)

[Objetivos 7](#_Toc104783363)

[Objetivo general 7](#_Toc104783364)

[Objetivos específicos 7](#_Toc104783365)

[Diseño de la Solución 8](#_Toc104783366)

[Base de datos del proyecto 8](#_Toc104783367)

[Procedimientos almacenados de la tabla alumno 8](#_Toc104783368)

[Tabla de contactos 10](#_Toc104783369)

[Procedimientos almacenados de la tabla contacto 11](#_Toc104783370)

[Tabla de notas 12](#_Toc104783371)

[Procedimientos almacenados de la tabla notas 12](#_Toc104783372)

[Procedimientos almacenados para consultar notas y contactos. 13](#_Toc104783373)

[Validaciones de los formularios 17](#_Toc104783374)

[¿Cómo generar el archivo .jar del proyecto? 17](#_Toc104783375)

[¿Cómo validar que los formularios no se repitan al abrirlos? 18](#_Toc104783376)

[Formularios del proyecto 19](#_Toc104783377)

[Login 19](#_Toc104783378)

[Registro 20](#_Toc104783379)

[Home 20](#_Toc104783380)

[Contactos 21](#_Toc104783381)

[Notas 21](#_Toc104783382)

[Conclusiones 22](#_Toc104783383)

[Recomendaciones 23](#_Toc104783384)

[Tabla de Ilustraciones 24](#_Toc104783385)

# **Introducción**

En el presente trabajo, se dejará constancia de cada uno de los pasos realizados para llevar a cabo nuestro proyecto, utilizando como lenguaje de programación Java y MySQL como lenguaje administrador de bases de datos. Dejando también evidencia sobre los respectivos diagramas que se utilizaron de base en cada paso a realizar en el proyecto.

En el trabajo, daremos a conocer sobre cómo está estructurado nuestro proyecto, implementando métodos eficaces para utilizar procedimientos almacenados, además, utilizaremos las mejores prácticas para dejar constancia de lo aprendido, también conoceremos sobre las diferentes formas o métodos que se utilizan para establecer una conexión a una base de datos utilizando el lenguaje de programación Java.

# **Problema Investigado**

Los alumnos de la universidad se muestran frente a una complejidad real al instante de realizar sus ocupaciones, tener resoluciones tecnológicas con la suficiente liviandad y efectividad a nivel privado en la organización; es una necesidad para lograr laborar ideas sin necesidad de tener que recurrir a plataformas externas generalmente que, en lugar de contribuir al enriquecimiento de los estudiantes, los distrae. La sociedad universitaria requiere de una plataforma donde logre guardar sus notas y sus contactos con mucha confianza; una plataforma con una interfaz amistosa donde la custodia de sus notas donde sea la máxima prioridad. Frente a lo propuesto, se muestra una solución al problema, para poder mejorar el rendimiento universitario a modo de un bloc personal de escrito plano con sus respectivos usuarios.

# **Justificación**

En un contexto social en el que la movilidad y la globalización son hechos cotidianos, cada vez, más se hace necesario pensar en herramientas que han sido útiles a lo largo de los años para que lo sigan siendo en este nuevo escenario en donde los puestos de trabajo, estudio o de ocio están allí, donde está esa conexión de internet, enfocándose en la idea donde la tecnología ha de servir para simplificar la vida cotidiana, con el tiempo se ha visto la necesidad de crear una herramienta tecnológica en el cual facilite al campo estudiantil logrando minimizar tiempo y ayudar a organizar y administrar actividades presentes y futuras por medio de un bloc de notas.

El presente proyecto se enfoca en la realización de un sistema en el cual consiste en mantener la simplicidad para aquellos estudiantes que solo buscan crear notas de forma rápida y sencilla, notas que pueden servir de base para otras herramientas más sofisticadas o que simplemente después de cumplir su papel son eliminadas, ofrece al campo estudiantil el manejo de información de actividades como: agregar apuntes y agregar contactos donde no habrá necesidad de utilizar diferentes aplicaciones ajenas a esta que muchas veces no brindan la confiabilidad necesaria.

La aportación principal de este proyecto será proveer de una vía alterna, sencilla y fiable para la simple gestión de notas personales o profesionales, brindado seguridad y confiabilidad a la información agregada a dicho bloc.

# **Objetivos**

## **Objetivo general**

Crear una herramienta tecnológica dirigida al sector estudiantil con el fin realizar diversas actividades como: agregar contactos, agendar actividades representadas en notas, toma de apuntes, en un espacio virtual la cual pretende tener un ambiente exclusivamente educativo con enfoque en la optimización de tiempo y mejor administración de actividades académicas.

## **Objetivos específicos**

1. Lograr un espacio estudiantil privado y exclusivo en el que los estudiantes se sientan seguros de guardar información privada de índole educativa
2. Diseñar un sistema fácil e intuitivo con una integración a una Base de Datos segura que será utilizado para y por los usuarios dentro del sector estudiantil.
3. Desplegar soluciones tecnológicas necesarias para poder integrar en el programa deseado los distintos componentes individuales dentro de la aplicación así logrando la funcionalidad deseada aplicando lo expuesto en clase para la necesidad.

# **Diseño de la Solución**

## **Base de datos del proyecto**

Como bien mencionamos, se realizo la base de datos en MySQL. En total son 3 tablas y 18 procedimientos almacenados, de igual forma, se utiliza la persistencia en cascada para insertar, eliminar y actualizar en las tablas.

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza media

Ilustración A. Tabla Alumno.

### Procedimientos almacenados de la tabla alumno

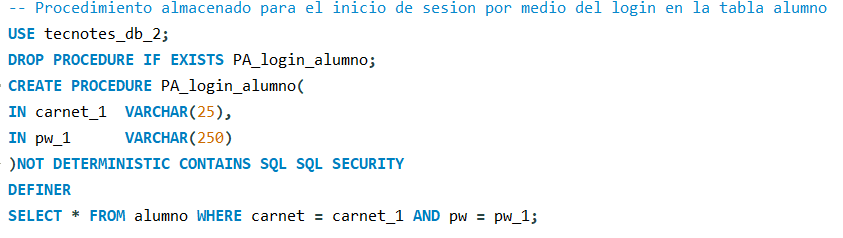


Ilustración B. Procedimiento almacenado de Login

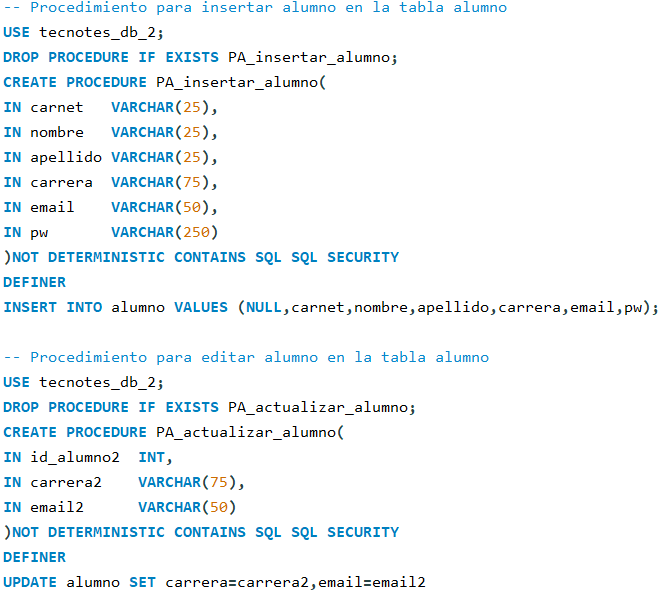


Ilustración C. Procedimientos almacenados para agregar

Texto

Descripción generada automáticamente con confianza baja

Ilustración D. Procedimiento almacenado para actualizar alumno

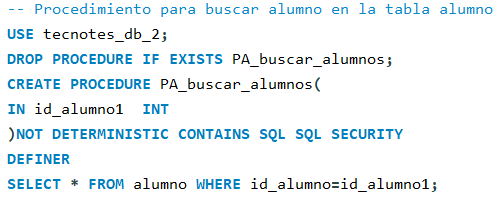


Ilustración E. Procedimiento almacenado para buscar alumno

Texto

Descripción generada automáticamente

Ilustración F. Procedimiento almacenado para eliminar alumno

### Tabla de contactos

Tabla

Descripción generada automáticamente

Ilustración G. Tabla contactos

### Procedimientos almacenados de la tabla contacto

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto, Correo electrónico

Descripción generada automáticamente

Ilustración H. Procedimiento almacenado para insertar contacto

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Ilustración I. Procedimiento almacenado para actualizar contacto

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

Ilustración J. Procedimiento almacenado para eliminar contacto

### Tabla de notas

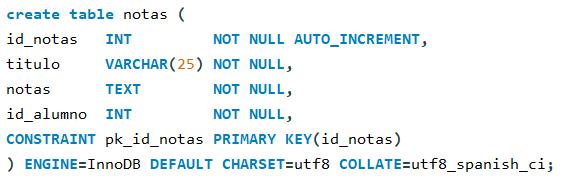


Ilustración K. Tabla de notas

### Procedimientos almacenados de la tabla notas

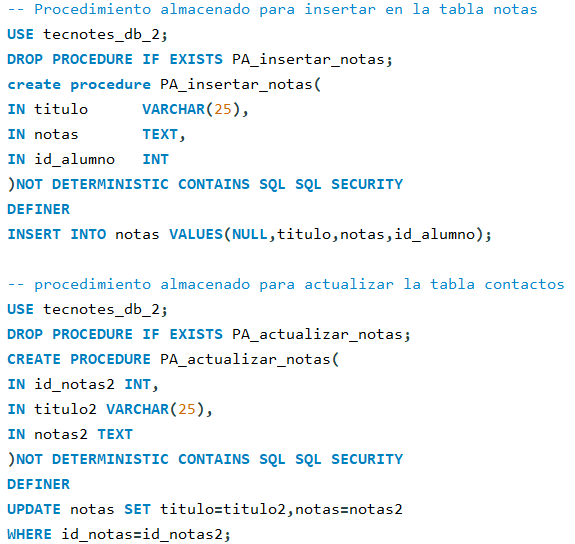


Ilustración L. Procedimiento almacenado para insertar nota

Tabla

Descripción generada automáticamente con confianza media

Ilustración M. Procedimiento almacenado para actualizar nota

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

Ilustración N. Procedimiento almacenado para eliminar una nota

### Procedimientos almacenados para consultar notas y contactos.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Ilustración O. Procedimiento almacenado para consultar los contactos de alumno

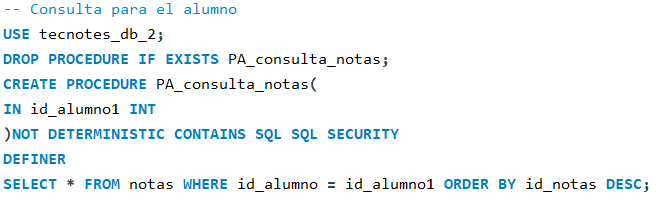


Ilustración P. Procedimiento almacenado para consultar las notas de alumno

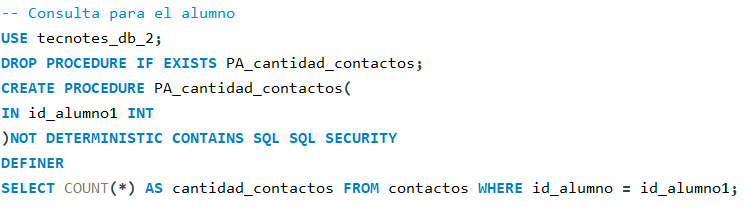


Ilustración Q. Procedimiento almacenado para obtener la cantidad de contactos de alumno

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Ilustración R. Procedimiento almacenado para obtener la cantidad de notas de alumno

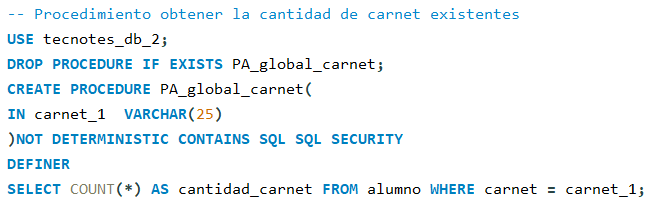


Ilustración S. Procedimiento almacenado para obtener la cantidad de carnet existentes

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

Ilustración T. Procedimiento almacenado para obtener la cantidad de emails existentes



Ilustración U. Procedimientos almacenados para procesar una nota de manera automática al estar registrado

Diagrama de la base de datos:



Ilustración V. Diagrama de la base de datos generado por MySQL

Diagrama entidad-relación:

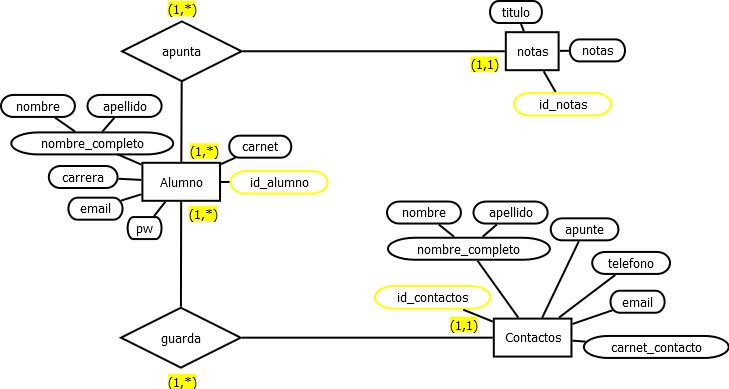


Ilustración 19 Diagrama Entidad-Relación

Diagrama relacional:

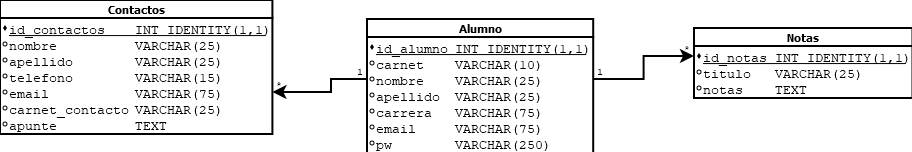


Ilustración 20. Diagrama Relacional

# Validaciones de los formularios

Las validaciones de los formularios en NetBeans fue una de las peticiones del proyecto por parte del docente entre ellas están:

### ¿Cómo generar el archivo .jar del proyecto?

Simplemente basta con darle clic derecho al proyecto en el que estemos trabajando y le damos a la opción que dice “Clean and Build”, esta acción tardará dependiendo del tamaño del proyecto y la velocidad del ordenador.

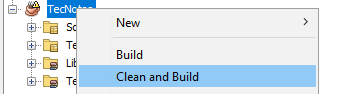


Ilustración W. Proceso del archivo .jar

Posteriormente, nos dirigimos a donde tenemos guardado nuestro proyecto, y nos movemos a la carpeta que dice “dist”, dentro de esa carpeta se encuentra nuestro archivo .jar

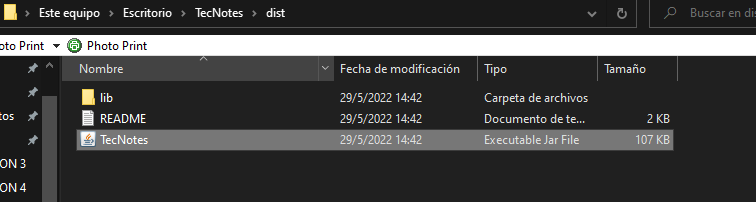


Ilustración 22 Dirección donde se guarda el archivo .jar

### ¿Cómo validar que los formularios no se repitan al abrirlos?

Nuestro equipo se puso de acuerdo en tratar de evadir complicaciones en código, por lo que optamos por la opción de añadir JDialogForm en vez de JFrameForm. Este formulario automáticamente hace la validación de no abrir varios formularios al presionar un botón, ya que es una ventana emergente, que no se cierra hasta cumplir su objetivo o hasta que el usuario decida salir.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Ilustración X. JDialogForm

Para agregar uno de estos formularios solo basta con agregarlo. Sin embargo, si no aparece entre las opciones, nos dirigimos a la opción que dice “Other” y dentro de ahí buscamos el JDialogForm.

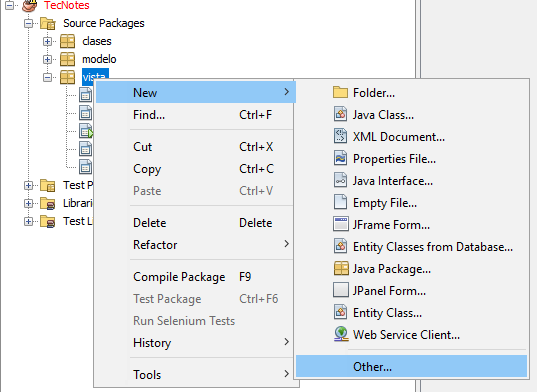


Ilustración Y. Agregar un JDialogForm

## **Formularios del proyecto**

El proyecto se desarrolló con el lenguaje de Java, y en el IDE de NetBeans. **Todos los formularios están validados con lo mencionado anteriormente.**

## Login

Ilustración 25 Login

Se realizo el formulario de registro para que los usuarios pudieran observar su respectiva información.

## Registro



Ilustración 26 Registro

Posteriormente al registrarse o ya tener la cuenta, se accede al formulario de Home.

## Home

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Ilustración Z. Home

Cuando el usuario presiona el botón de “Mis contactos”, lo redirige al formulario de contactos.

## Contactos

Interfaz de usuario gráfica

Descripción generada automáticamente

Ilustración AA. Contactos

De igual forma, si el usuario presiona el botón de “Mis notas”, lo redirige al formulario de notas.

## Notas

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Ilustración BB. Notas

# **Conclusiones**

En conclusión, la herramienta facilita al campo estudiantil ayudando a organizar y administrar actividades presentes y futuras por medio de un bloc de notas; como agendar notas y agregar personas a la lista de contactos y actividades académicas asignadas por los docentes creando así un ambiente exclusivo

Dando las soluciones necesarias para la aplicación cumpliendo con los objetivos planteados en el documento. Manteniendo la simplicidad para aquellos que solo buscan crear notas de forma rápida.

# **Recomendaciones**

Establecer un lugar específico para universidad y notaciones de clases ahora es posible, somos la alternativa, al utilizar aplicaciones de terceros en las que se puede distraer y/o descontinuarse de los que se está realizando.

* Utilizar nuestra aplicación, es segura y compatible con la plataforma Windows, donde puedes registrarte para tener únicamente acceso a todas tus notas o contactos. Tienes la facilidad de plasmar tu información estudiantil.
* En el apartado de contactos debes guardar los datos de tus amigos y/o compañeros de universidad con las que compartirás en la asignatura cursante.
* En el apartado de las notas, ya que en ellas se plasman todas las notaciones sobresalientes, apuntes, pendientes, importantes y notas como tal, con la posibilidad de integrar la fecha de dicha nota.
* Lo que se busca es: tener la mejor accesibilidad de parte de los estudiantes, para que se pueda llevar orden y documentación de la experiencia universitaria.

# **Tabla de Ilustraciones**

[Ilustración A. Tabla Alumno. 8](#_Toc104783279)

[Ilustración B. Procedimiento almacenado de Login 8](#_Toc104783280)

[Ilustración C. Procedimientos almacenados para agregar 9](#_Toc104783281)

[Ilustración D. Procedimiento almacenado para actualizar alumno 9](#_Toc104783282)

[Ilustración E. Procedimiento almacenado para buscar alumno 10](#_Toc104783283)

[Ilustración F. Procedimiento almacenado para eliminar alumno 10](#_Toc104783284)

[Ilustración G. Tabla contactos 10](#_Toc104783285)

[Ilustración H. Procedimiento almacenado para insertar contacto 11](#_Toc104783286)

[Ilustración I. Procedimiento almacenado para actualizar contacto 11](#_Toc104783287)

[Ilustración J. Procedimiento almacenado para eliminar contacto 12](#_Toc104783288)

[Ilustración K. Tabla de notas 12](#_Toc104783289)

[Ilustración L. Procedimiento almacenado para insertar nota 12](#_Toc104783290)

[Ilustración M. Procedimiento almacenado para actualizar nota 13](#_Toc104783291)

[Ilustración N. Procedimiento almacenado para eliminar una nota 13](#_Toc104783292)

[Ilustración O. Procedimiento almacenado para consultar los contactos de alumno 13](#_Toc104783293)

[Ilustración P. Procedimiento almacenado para consultar las notas de alumno 14](#_Toc104783294)

[Ilustración Q. Procedimiento almacenado para obtener la cantidad de contactos de alumno 14](#_Toc104783295)

[Ilustración R. Procedimiento almacenado para obtener la cantidad de notas de alumno 14](#_Toc104783296)

[Ilustración S. Procedimiento almacenado para obtener la cantidad de carnet existentes 15](#_Toc104783297)

[Ilustración T. Procedimiento almacenado para obtener la cantidad de emails existentes 15](#_Toc104783298)

[Ilustración U. Procedimientos almacenados para procesar una nota de manera automática al estar registrado 15](#_Toc104783299)

[Ilustración V. Diagrama de la base de datos generado por MySQL 16](#_Toc104783300)

[Ilustración W. Proceso del archivo .jar 17](#_Toc104783301)

[Ilustración X. JDialogForm 18](#_Toc104783302)

[Ilustración Y. Agregar un JDialogForm 19](#_Toc104783303)

[Ilustración Z. Home 20](#_Toc104783304)

[Ilustración AA. Contactos 21](#_Toc104783305)

[Ilustración BB. Notas 21](#_Toc104783306)