

Instituto de Educación Superior Tecnológico Público

"De las Fuerzas Armadas"



TRABAJO DE APLICACIÓN PROFESIONAL

**IMPLEMENTACIÓN DE UNA APLICACIÓN WEB PARA LA
GESTIÓN ACADÉMICA DE I.E VICTORIA BARCIA BONIFFATTI –
SAN JUAN DE LURIGANCHO**

**PARA OPTAR EL TÍTULO PROFESIONAL TÉCNICO EN
ANÁLISIS DE SISTEMAS**

PRESENTADO POR:

CCAMA HUERTA, Victor Jesus

DAVILA ORTIZ, Erick Humberto

RAMOS CHAVEZ, Josue Josias

LIMA, PERÚ

2020

A la Institución Educativa Victoria Barcia Boniffatti, por darnos la oportunidad de desarrollar y aplicar cada uno de nuestros conocimientos y por creer en nuestras capacidades, este trabajo de aplicación también está dedicado a Dios por darnos la vida; a nuestros padres, familiares por su apoyo incondicional; a nuestros docentes , por su paciencia, por compartir su sabiduría y experiencia en el transcurso de nuestra formación técnica.

AGRADECIMIENTOS

A todos los que nos han ayudado a completar una etapa de nuestra vida, hacer realidad nuestros sueños y ver los resultados de grandes esfuerzos, por esto, nos gustaría agradecer todo el trabajo en este proyecto y aplicación que se puede completar con éxito.

A nuestros padres por ser los principales motivos de nuestros sueños, por darnos la confianza de y fortaleza en los momentos que mas necesitamos.

A nuestros docentes, por habernos compartido sus conocimientos a lo largo de la preparación de nuestra profesión.

ÍNDICE

	Página
Resumen	viii
Introducción	ix
CAPÍTULO I. DETERMINACIÓN DEL PROBLEMA	11
1.1 Formulación del problema	12
1.1.1. Problema general	12
1.1.2. Problema específicos	12
1.2 Objetivos	12
1.2.1 Objetivo general	12
1.2.2 Objetivos específicos	13
1.3 Justificación	13
CAPÍTULO II. MARCO TEÓRICO	14
2.1 Estado de arte	15
2.2 Bases teóricas	17
2.2.1 Aplicación Web	17
2.2.2 Arquitectura Cliente/Servidor	18
2.2.3 Herramientas	18
CAPÍTULO III. DESARROLLO DEL TRABAJO	25
3.1 Finalidad	26
3.2 Propósito	26
3.3 Componentes	26
3.4 Actividades	27
3.5 Limitaciones	112
CAPÍTULO IV. RESULTADOS	113
4.1 Resultados	114
CAPÍTULO V. CONCLUSIONES Y RECOMENDACIONES	115
5.1 Conclusiones	116
5.2 Recomendaciones	117
Referencias	118
APÉNDICES	
Apéndice A. Cronograma de Actividades	
Apéndice B. Cronograma de Presupuesto	
Apéndice C. Manual de usuario	

LISTA DE FIGURAS

	Página
Figura 1. Aplicaciones Web	17
Figura 2. Arquitectura Cliente/Servidor	18
Figura 3. PHP Hypertext Preprocessor	19
Figura 4. MySQL	20
Figura 5. JavaScript	21
Figura 6. Bootstrap	21
Figura 7. JQuery	22
Figura 8 . Asynchronous Javascript Anda XML – AJAX	22
Figura 9. Modelo MVC	23
Figura 10 . Erwin Data Modeler	24
Figura 11. Etapa de Desarrollo de Trabajo	27
Figura 12. Diagrama de caso de uso Registro de asistencia(actual)	28
Figura 13. Diagrama de caso de uso Proceso Horario (actual)	29
Figura 14. Diagrama de caso de uso Registro de Matrícula (actual)	29
Figura 15. Diagrama de caso de uso Registro de Notas (actual)	30
Figura 16. Diagrama de caso de uso Registro de asistencia (Propuesto)	31
Figura 17. Diagrama de caso de uso Registro de Horario (Propuesto)	31
Figura 18. Diagrama de caso de uso Registro de Matricula(Propuesto)	32
Figura 19. Diagrama de caso de uso Registro de Notas (Propuesto)	32
Figura 20. Estructura de la pagina principal (Extranet)	33
Figura 21. Esquema general del sitio web	34
Figura 22. Maquetacion de Pagina Principal	34
Figura 23. Estructura del Sistemas	35
Figura 24. Diagrama Entidad Relación	36
Figura 25. Inicio de Sesión del Administrador	99
Figura 26. Inicio Para registro de Matrícula	100
Figura 27. Registro de Matricula	101
Figura 28. Inicio de Sesion del Docente	101
Figura 29. Menú Docente	102
Figura 30. Boton de cerrar sesión	102
Figura 31. Cierre de sesión	103

	Página
Figura 32.Botones de Acceso	103
Figura 33.Relacion de Estudiantes	103
Figura 34.Tablero de Notas	104
Figura 35.Tablero de Horarios	104
Figura 36.Inicio de Sesion Estudiante	105
Figura 37.Menu Estudiante	106
Figura 38.Boton Cerrar Sesión	106
Figura 39.Cerrar Sesión	106
Figura 40.Botones de acceso	107
Figura 41.Tabla de Notas	107
Figura 42.Reporte de Notas	107
Figura 43.Detalles de Notas	108
Figura 44.Programacion de Horario	108
Figura 45.Inicio al perfil de docente	108
Figura 46.Pagina Principal Estudiante	109
Figura 47.Boton de Cierre de Sesión	110
Figura 48.Cerrar Sesión	110
Figura 49.Botones de acceso	110
Figura 50.Reporte de Notas	110
Figura 51.Registro de Notas	111
Figura 52.Detalles de Notas	111
Figura 53.Reporte de Horario	112

RESUMEN

El presente trabajo de aplicación profesional se realizó con la finalidad de satisfacer y mejorar los procesos de Gestión Académica de la Institución Educativa Victoria Barcia Boniffatti, ubicada en el Jr. Las Chulpas de la Urbanización Zarate del distrito de San Juan de Lurigancho. Lima – Perú.

La situación actual del colegio es que los procesos de registros de notas, registros de asistencias, así como también la programación de los ambientes (aulas y laboratorio de cómputo) se realizan de forma manual en fichas preimpresas y en muchos casos en hojas de cálculo como el Excel.

Este trabajo consiste básicamente en el desarrollo de una Aplicación web que interactúa con una base de datos la cual ayudara a mejorar dicho proceso.

Estos procesos que actualmente existen han generado mucha insatisfacción por parte del personal administrativo y docentes, debido a que la información y documentos no está organizada de forma sistematizada, generando confusión entre los profesores, al momento de desplazarse a sus aulas de clases, al no tener un control adecuado tanto en la Programación de Clases, registro de notas y control de asistencia de los estudiantes. Generándose pérdida de tiempo en las labores académicas y administrativa, retrasando así la gestión del control de asistencia, actas de notas, así como también en el tiempo que le toma a cada docente entre la ubicación y desplazamiento a su aula respectiva.

Toda la documentación se encuentra almacenada en múltiples archivadores de forma tradicional en formatos preimpresos, registros a lápiz y archivos Excel almacenados en las diferentes computadoras de la institución. Respecto al financiamiento del presente trabajo de investigación básicamente fueron los costos del recurso humano que participó en la construcción del sistema y los costos de licenciamiento son nulos puesto que se utilizó el software libre.

Palabras claves: Implementación, aplicación web, gestión académica, gestión educativa,

INTRODUCCIÓN

El proyecto es el resultado de una serie de coordinaciones con la dirección de la Institucion de Educación Victoria Barcia Boniffatti ubicado en el distrito de San Juan de Lurigancho. Lima, Perú.

El objetivo principal del proyecto es la mejora de los siguientes procesos; registro de datos generales del alumno, registro de salones, Registro de matrícula, Registro de Notas, Registro de Asistencia, Emision del reporte de alumnos, Emision del reporte de notas.

La base de datos que se utiliza en este proyecto será MYSQL y el IDE de desarrollo es NETBEANS usando el lenguaje de programación PHP. La data será almacenada en este servidor de base de datos donde podrá ser consultada en tiempo real vía web desde cualquier computadora con internet. Asimismo, el sistema puede ser utilizado desde cualquier dispositivo móviles.

Con esta aplicación web se mejorará significativamente la gestión académica en el colegio. Principalmente el proceso de registros de notas, registro de horario, registro de matrícula, registros de asistencias, así como también la programación de los ambientes (aulas y laboratorio de cómputo). Asimismo, se podrán generar reportes estadísticos para mejorar el proceso de toma de deciones de los directivos.

Para el desarrollo de este trabajo se utilizaron las siguientes herramientas; Microsoft Project (Planificacion del proyecto), Rational Rose (Análisis y Diseño del Sistema), ERWIN (Modelamiento de Base de Datos), MYSQL (Motor de Base de Datos), PHP (Lenguaje de Promacion), APACHE SERVER (Servidor Web), NETBEANS (IDE de desarrollo), BOOTSTRAP (Framework de Interfaz), FPDF (Framework de Reportes), JQUERY(Libreria de JavaScript). Finalmente, se puede indicar que el objetivo principal de este proyecto es mejorar la Gestion académica del colegio así como la imagen de la institución.

En el capítulo I, se explica la forma como identificaron los problemas de la Insitución.

En el capítulo II, se recopila información de proyectos o trabajos de investigación semejantes a este proyecto. Con objeto de tener amplia información para elaborar un diseño o propuesta óptima.

En el capítulo III, se analiza la problemática de la institución educativa a través de encuestas y reuniones realizadas con el personal directivo de la institución. Asimismo, se explica la función de la herramienta Rational Rose para la etapa de modelamiento de procesos actuales y propuestos. También se detalla el paso a paso la implementación de la pagina web en la que se mencionan las herramientas utilizadas como PHP, Javascript, html, MySQL, CSS.

En el capítulo IV, se detallan los resultados obtenidos en función a los objetivos específicos.

En el capítulo V, en este capítulo se procede a desarrollar y determinar los puntos que forman la última fase que hemos alcanzado.

CAPÍTULO I

DETERMINACIÓN DEL PROBLEMA

1.1. Formulacion del problema

En la Institucion Educativa Victoria Barcia Boniffatti del distro de San Juan de Lurigando, se detectó la falta de automatización en el manejo del los procesos dentro de la gestión académica en donde se realizaban los registros de actas de notas, registro de asistencia, registro de matrícula y las designaciones de aulas y laboratorios en hojas fisicas y a través del uso de la herramienta exel, es así como nos dimos cuenta que la gestión académica de la institución tendría que tener una mejora.

Por ello, con la implementación de el aplicativo web se mejorará la gestión académica de la institución ya que brindará un mejor servicio y facilidad para que los procesos mencionados tengan una mejor flexibilidad debido al menejo de información que se trabaja en los registros

1.1.1 Problema general

1.0 ¿De qué manera una Aplicacion Web mejora la gestión académica de la Institución Educativa “Victoria Barcia Boniffatti”?

1.1.2 Problemas específicos

- 1.1 ¿De qué manera una aplicación web mejora el proceso de registro de actas de notas en la Institución Educativa Victoria Barcia Boniffatti?
- 1.2 ¿De qué manera una aplicación web mejora el proceso de registro de asistencia de alumnos en la Institución Educativa Victoria Barcia Boniffatti?
- 1.3 ¿De qué manera una aplicación web mejora el proceso de registro de matrícula en la Institución Educativa Victoria Barcia Boniffatti?
- 1.4 ¿De qué manera una aplicación web mejora el proceso de distribución de aulas y laboratorios en la Institución Educativa Victoria Barcia Boniffatti ?

1.2. Objetivos

1.2.1 Objetivo general

- 1.0 Implementar un Sistema Web para agilizar la gestión académica de la Institución Educativa “Victoria Barcia Boniffatti”

1.2.2 Objetivos específicos

- 1.1 Implementar una aplicación web que permita mejorar el proceso de registro de actas de notas en la Institución Educativa Victoria Barcia Boniffatti.
- 1.2 Implementar una aplicación web que permita mejorar el proceso de registro de asistencia de alumnos en la Institución Educativa Victoria Barcia Boniffatti.
- 1.3 Implementar una aplicación web que permita mejorar el proceso de registro de matrícula en la Institución Educativa Victoria Barcia Boniffatti.
- 1.4 Implementar una aplicación web que permita mejorar el proceso de distribución de aulas y laboratorios en la Institución Educativa Victoria Barcia Boniffatti.

1.3 Justificación

Al realizar un diagnóstico en la Institución Educativa detectamos la necesidad que requería a través de esta implementación web, en donde se mejorará el proceso de gestión académica de manera automatizada. La propuesta planteada permitirá ayudar a difundir y mejorar el control de los siguientes registros de notas, asistencia, matrícula como también la distribución de laboratorios y aulas.

Los beneficios que se obtendrán con la implementación del aplicativo web serán:

- a) Un adecuado control en el registro de las actas de notas de los alumnos, el cual permitirá a cada apoderado mostrar la información correspondiente a las notas de su hijo.
- b) Un adecuado control en el registro de la asistencia de los alumnos, el cual permitirá a cada apoderado mostrar la información correspondiente a las asistencias de su hijo.
- c) Un adecuado control en el registro de matrículas de notas de los alumnos, el cual permitirá a cada apoderado mostrar la información correspondiente a su estado de pagos y la nómina de matrícula de su hijo.
- d) Una adecuada distribución de las aulas y los laboratorios, el cual permitirá a cada docente y alumno poder visualizar sus asignaciones correspondientes.

CAPÍTULO II
MARCO TEÓRICO

2.1 Estado de Arte

En el desarrollo de una investigación objetiva que guarde una línea direccional en relación con otros trabajos desarrollados bajo el mismo tema, se realizó una revisión a diferentes investigaciones el cual nos ayudó y sirvió como referencias de tal manera que incluyendo trabajos de cualquier fuente de información educativa alusiva al tema de estudio se muestran investigaciones en las cuales se uso una aplicación Web como herramienta de comunicación e información.

Antecedentes de estudio

Huaman (2019) en su trabajo de investigación “Influencia de una Aplicación móvil en el proceso de Matrícula y Consulta académica en los estudiantes de la facultad de ingeniería de la Universidad Privada Antonio Guillermo Urrelo”, El objetivo aquí mencionado fue crear un portal para que estudiantes y administradores utilicen sus dispositivos móviles personales para realizar actividades, de manera que pueda convertirse en una herramienta de apoyo a la práctica docente y mejorar el proceso docente de profesores y personal. La evaluación del aprendizaje se basó en la teoría constructivista, y se adopta un modelo de proyecto factible en estudios de campo no experimentales, porque la dirección del trabajo es responder a las necesidades de información escolar que brinda la Facultad de Ingeniería. Fue propuesto por la Universidad Privada Antonio Guillermo Urrelo y planteó una especie de adopción La página web es un método para difundir información escolar de forma rápida, eficaz y estimulante.

Marquez (2018) en la tesis “ Implementacion de un sistema informatico de matrículas y pensiones de la I.E.P. Santa Isabel de Hungria – Casma ”, tuvo como objetivo implementar un sistema computarizado de inscripción y pensión para I.E.P. Saint Elisabeth-Kasma, Hungría; en 2018, para mejorar la calidad administrativa del procesamiento de la información; el estudio utiliza un diseño no experimental, descriptivo y transversal, y la población y la muestra se limitan a 40 personas. Para la recolección de datos se utilizó una herramienta de encuesta mediante cuestionario a través de la tecnología de encuestas, la herramienta de encuesta obtuvo los siguientes resultados: 97,50% de la población encuestada creía que se debía instalar un sistema informático. Estos resultados son consistentes con los supuestos generales, por lo que es necesario implementar un sistema informático para ayudar a mejorar a Santa Isabel de I.E.P. Hungría.

Uribe (2017) en su investigación “Propuesta del sistema web para la gestión de Matrícula y Registro de Notas del nivel Secundario del Colegio Privado Peruano Americano – Huaraz ” tuvo como propósito implementar un sistema de red para mejorar la gestión académica de matrículas y calificaciones; se propuso crear un módulo de reporte y verificación en la información a través de una base de datos. Su objetivo fue controlar y administrar los registros académicos en en la Institución Educativa Privada Peruano Americano. Para su estudio utilizó un diseño no experimental, documentado y descriptivo, con una población de 29 trabajadores y una muestra limitada de 29 usuarios. En cuanto a los siguientes aspectos, se utilizaron medios medibles. Se pudo determinar que el 93% de las personas dice que las instituciones educativas requieren lo siguiente: El 7% de los sistemas de gestión de registros y calificaciones consideran innecesario implementarlo. Todos estos resultados son consistentes con los supuestos generales, por lo que la razón razonable de este estudio es la necesidad de implementar un sistema de información que ayude a mejorar la gestión de matrículas y los procesos de gestión de registros de calificaciones de las Institución Privada Peruana Americana Huaraz 2017.

Ayauja y Delgado (2017) en la tesis “Estudio y Diseño de una Aplicación informática para el control de Matrícula de La Facultad de Agronomía de la Universidad Nacional San Luis Gonzaga de Ica” tuvo como propósito contribuir a la Facultad de Ingeniería Agronómica para que un sistema de información pueda ordenar las unidades de admisión, razón por el cual se creó el sistema informático, para ello se utiliza una metodología orientada a objetos. Un lenguaje de modelado unificado, que nos permite dibujar diagramas de procesos. De manera similar, Rational Rose se utiliza para modelar el proceso y crear la base de datos necesaria para controlar la información. Una vez finalizado el proyecto de tesis se podrán alcanzar todas las metas planteadas, de esta forma el trabajo actual se convertirá en una herramienta importante para el desarrollo de las actividades académicas de la unidad de matrícula, y aquí se procesará toda la información.

Bernardo y Paredes (2016) en la tesis “Aplicación de la Metodología Six Sigma para mejorar el proceso de registros de Matrícula en la Universidad Autónoma del Perú ”, propusieron mejorar el proceso de registro de matrícula de la Universidad Autónoma del Perú, esta investigación está enmarcada mediante la modalidad de proyecto factible apoyada en una metodología Six Sagma, que ayudó, utilizando como estrategia, mejorar el proceso centrado en la reducción de la variabilidad de los mismo reforzando y optimizando cada parte de el proceso y consiguiendo reducir o eliminar los defectos o fallos en la entrada del producto o servicio.

En donde se ve concretamente la observación documental y la de campo, esta última se logró mediante la aplicación de un instrumento de recolección de datos e información constituido por un cuestionario al que se le aplicó el método de escalamiento de Likert. Los resultados obtenidos fueron tabulados, graficados y analizados con el objeto de obtener la evidencia de la necesidad existente de información que poseen los estudiantes de la la Universidad Autónoma que podrá ser cubierta a través del uso de las nuevas tecnologías de la comunicación y la información como lo constituye la internet a través de la propuesta de mejorar el proceso de matrícula usando la metodología Six Sigma. Esta investigación se correlaciona con el presente trabajo de investigación, de acuerdo a que ambas investigaciones tienen el objetivo de diseñar una página web que sirva de herramienta informativa y educativa, respecto a la actividad académica de la institución educativa en particular.

2.2 Bases teóricas

2.2.1 Aplicación Web

Ramos y Ramos (2014) definen que las Aplicaciones Web “son sistemas informáticos que están compuestos de software y hardware y que atienden y responden a diversas peticiones de los navegadores Web” (p.28).

Entonces, se infiere que las aplicaciones web brindarán el apoyo automatizando procesos con el propósito de mejorar las demandas del factor tiempo que se encuentra en gestiones dando así la facilidad de que usando las aplicación web se pueden atender diversas peticiones para cualquier uso.



Figura 1. Aplicaciones Web 2.0 (Noguera, 2018)

2.2.2 Arquitectura Cliente/Servidor.

Luján-Mora (2002) sostiene que las arquitecturas Cliente/servidor son sistemas que tienen la característica principal que operan en dos capas. Que son la capa Cliente y la capa Servidor. La capa servidor esta compuesta de un equipo de computo que opera con un sistema operativo de servidor que controla todos los equipos de la y un servidor de base de datos que almacena toda la data de la empresa. La capa cliente esta compuesta de un equipo que opera con un sistema operativo de cliente y que tiene instalado la aplicación informática que se conecta con el servidor de base de datos.

Entonces, se infiere que la arquitectura tradicional de cliente-servidor también es conocida como arquitectura de dos capas. Donde se encuentran los múltiples recursos y aplicaciones por el lado del servidor y que los pone a disposición de los clientes cada vez que éstos lo requieran.

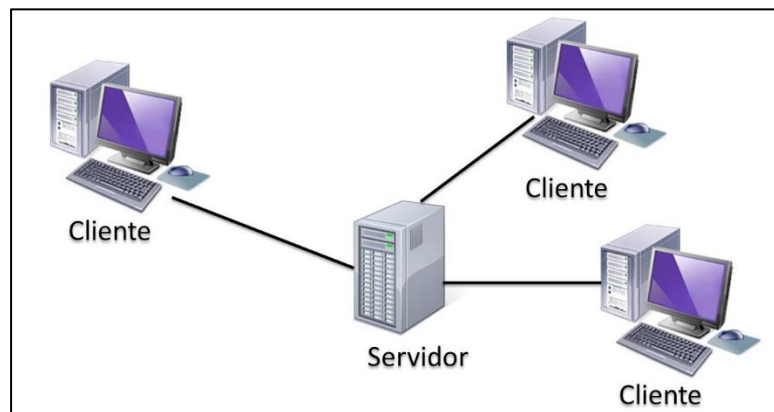


Figura 2. Arquitectura Cliente/Servidor (Huertas, 2019)

2.3 Herramientas

2.3.1 PHP (Acrónimo de "PHP: Hypertext Preprocessor")

Es un lenguaje de "código abierto" interpretado, de alto nivel, embebido en páginas HTML y ejecutado en el servidor. Según Granado (2009) PHP es un lenguaje de programación que crear páginas web dinámicas, esto implica la utilización de varios conjuntos de código compatible con html para un performance acorde a los requerimientos.



Figura 3. PHP Hypertext Preprocessor (Aguirre, 2020)

2.3.2 MySQL

Spona (2010) sostiene que es un sistema de administración de bases de datos porque es de libre acceso, por lo que entra en la categoría de programas de código abierto porque no tiene el costo de un sistema propietario. Según (McLaughlin), ha sido ampliamente utilizado en aplicaciones de entorno Web, Según el método utilizado en el diseño del sistema, se relaciona directamente con PHP en el proceso de desarrollo, sin subestimar las capacidades de otros administradores de bases de datos. Cuando MySQL usa un motor de almacenamiento sin transacciones, MySQL lee muy rápido En las aplicaciones web, el entorno requiere una lectura de datos intensiva, lo que hace que MySQL sea muy adecuado para tales aplicaciones.



Figura 4. MySQL (Gómez, 2015)

2.3.3 Java Script

Flanagan (2007) afirma que Java Script es un lenguaje de programación interpretado y un dialecto del estándar ECMAScript. Se define como orientado a objetos, basado en prototipos, imperativo, de tipo flexible y dinámico. Se utiliza principalmente en forma de cliente, implementado como parte de un navegador web, que puede mejorar la interfaz de usuario y las páginas web dinámicas, a pesar de la presencia de JavaScript del lado del servidor (JavaScript del lado del servidor) o SSJS). También es importante su uso en aplicaciones externas en la web (como documentos PDF) y aplicaciones de escritorio (principalmente widgets).

Desde 2012, todos los navegadores modernos son totalmente compatibles con ECMAScript 5.1 (una versión de JavaScript). Los navegadores más antiguos admiten al menos ECMAScript3. La sexta edición se publicó en julio de 2015. La sintaxis de JavaScript es similar a C, aunque usa los nombres y convenciones del lenguaje de programación Java. Sin embargo, Java y JavaScript tienen diferentes semánticas y propósitos. Todos los navegadores modernos interpretan el código JavaScript incrustado en las páginas web. Para interactuar con las páginas web, se proporciona el lenguaje JavaScript con la implementación del Modelo de Objetos de Documento (DOM). Tradicionalmente, se ha utilizado para realizar operaciones en páginas web HTML y solo realizar operaciones dentro del marco de una aplicación cliente, sin acceso a las funciones del servidor.

Según lo anterior, javascript es un lenguaje de programación orientado a objetos principalmente del lado del cliente, que permite la creación de nuevos contenidos. Implementado como parte de un navegador web, puede mejorar la interfaz de usuario y las páginas web dinámicas.



Figura 5. JavaScript (Montoya, 2019)

2.3.4 Bootstrap

“Es una biblioteca multiplataforma o conjunto de herramientas de código abierto utilizado en el diseño de aplicaciones web, mediante el uso de plantillas de diseño con botones, cuadros, menú de navegación y otros elementos de diseño basados en HTML y CSS “(Hernández, 2020, p.15).



Figura 6. Bootstrap (Colimba 2016, p, 28)

2.3.5 AJAX

Arias (2014) menciona que para lograr la conectividad sin sincronizar con el servidor, proporcionando así a los usuarios una mayor interactividad con el mapa, se utiliza un tipo bastante nuevo llamado AJAX. Este no es una tecnología nueva, sino una combinación de múltiples tecnologías y estándares: Javascript como lenguaje de programación, XML para intercambio de datos, DOM para procesar objetos de página y Modelo de comunicación asíncrona para intercambio de datos. Se ejecutan en el navegador del usuario y mantienen una comunicación asincrónica con el servidor en segundo plano.



Figura 7. Asynchronous Javascript And XML – AJAX
(Instituto FOC, 2016)

2.3.6 JQuery

Zeas (2015) define a “jQuery es una librería de JavaScript, es software libre y de código abierto. Permite interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones, utilizar AJAX en las páginas web, simple de usar y es multiplataforma” (p. 14).

Entonces Se define que JQuery es una librería de JavaScript que permite realizar interactividad a una página web, ofrece una serie de funciones y métodos con los cuales facilita al usuario para construir una página web de manera rápida y sencilla.



Figura 8 . JQuery (Cumbo, 2016)-

2.3.7 MVC (Modelo-Vista-Controlador)

Según Bahit (2011) “El modo MVC es un patrón de arquitectura de software, responsable de separar la lógica empresarial de la interfaz de usuario, y es el modo más utilizado en aplicaciones web, ya que promueve la función, mantenibilidad y escalabilidad del sistema de una manera simple y fácil de usar. Para lograr el siguiente propósito no se permite mezclar lenguajes de programación en el mismo código” (p. 36). MVC divide en 3:

- Modelo: Es responsable del acceso directo a los datos y actúa como "intermediario" de la base de datos.
- Vista: Responsable de mostrar información a los usuarios de forma gráfica y "fácil de leer".
- Controlador: Es un intermediario entre la vista y el modelo. Controla la forma en que interactúa con el usuario solicitando datos del modelo y entregándolos para su visualización, de modo que puedan presentarse al usuario de una manera "legible por humanos".

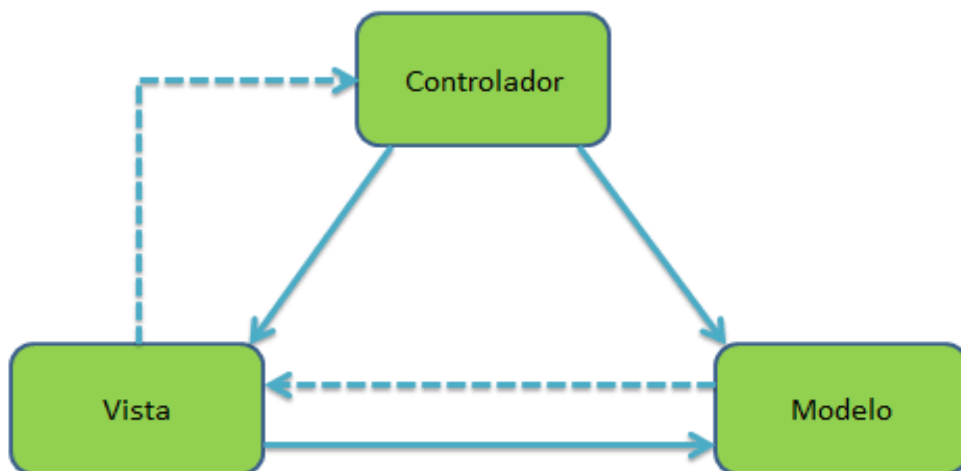


Figura 9. Modelo MVC (Blancarte, 2014)

2.3.8 Erwin Data Modeler

Burbank, y Hoberman (2011) sostienen que el modelado de datos en la industria brinda la interfaz simple y visual para administrar cual quier complejo en entorno de datos en la edición Standard ya que es una versión de modelado para equipos de escritorio individuales.

De acuerdo a la concepción anterior se infiere que Erwin es una herramienta orientado en su totalidad al modelado de bases de datos relacional, es flexible y nos proporciona desde un modelo lógico de los requerimientos de información hasta el modelo físico.



Figura 10 . Erwin Data Modeler (A.Adivina, 2016)

CAPÍTULO III
DESARROLLO DE TRABAJO

3.1 Finalidad

La finalidad del proyecto de tesis es la gestión académica en la Institución Educativa en donde los registros de matrícula, registro de asistencias, registro de actas de notas, organización y distribución de aulas y laboratorios, son registro que toda entidad educativa, esto permitirá mejorar la gestión académica de la Institución Educativa y brindará una mayor difusión y almacenamiento de la información de los reportes de notas, asistencias, matrículas y la elaboración de los horarios para el bienestras de la institución y alumnos.

3.2 Propósito

El propósito de este proyecto de tesis es implementar un aplicativo web dinámica que interactue con una base de datos para resolver los problemas de perdidas de registros de documentación con información valiosa también mejorara la difusión entre la institución educativa y sus usuarios.

3.3 Componentes

- a) **Erwin Data Modeler** : Se utilizo esta herramienta para modelar toda la estructura de la base de datos.
- b) **MySQL** : Este gestor se utilizo para realizar toda la programación SQL.
- c) **Netbeans**: Este entorno de desarrollo se utilizo para realizar la programación orientada a obeitos en el lenguaje PHP.
- d) **PHP**: Este lenguaje se utilizo para realizar la programación de construir toda la aplicación web .
- e) **Bootstrap**: este framework se utizo para diseñar toda la interfás de la aplicación web.
- f) **JavaScript**: Este lenguaje de programación se utilizo para poder dar interactividad a la apliacion web.
- g) **Jquery**: esta librería nos sirvió para complementar la programcion de JavaScript.
- h) **Ajax**: permitió la comunicacion rápida con el servidor generando respuestas a las peticiones que el cliente realiza.
- i) **HTML**: Este lenguaje se utilizo para realizar la estructura de la aplicación web.

3.4 Actividades

En el desarrollo de este proyecto realizamos las siguientes etapas la cual iniciamos con la recopilación de la información, seguidamente el análisis de la información, luego el diseño del sitio web, después la codificación del sitio web y finalmente las pruebas del sitio web. Estas son las etapas y tareas que corresponden a una metodología secuencial.



Figura 11. Etapa de Desarrollo de Trabajo

Recopilación de la información

Para recopilar la información de la institución educativa tuvimos que utilizar una encuesta, la cual consistió en extraer información mediante un formulario de preguntas que fue llenado por cada docente, personal administrativo, padre de familia o apoderado y alumnos en general. usando la encuesta como instrumento de estudio pudimos recolectar la información necesaria.

Elaboracion de Encuesta al Personal administrativo, docente, padres de familia y alumnos de la Institución Educativa Victoria Barcia Boniffatti

- 1- ¿Usted se encuentra satisfecho con la difusión de los comunicados de reportes de los registros que realiza la institución educativa?
- 2- ¿Por qué cree usted que la implementación de un sitio ayudara a mejorar la gestión académica de la institución educativa ?
- 3- ¿Para usted que medio de comunicación utiliza para informarse de las notas, asistencias y matrículas de sus hijos?

- 4- ¿Qué tipos de contenidos desea que se muestren en la pagina web?
- 5- ¿Cree que se puede mejorar los servicios educativos de la institución educativa mediante la implementación de un sistema?
- 6- ¿Usted cree que es necesario implementar un aplicativo web para que la institución educativa brinde información ?

Análisis de la Información

Análisis del sistema Actual.- Para tener la información recopilada en la etapa de levantamiento de información, tuvimos que elaborar diagramas de caso de uso la cual demuestran la problemática de la institución educativa. Aquí se pudo observar los problemas principales de la institución educativa que son:

- Registro de notas en excel
- Registro asistencia en excel
- Registro de matrícula en excel
- Designación de aulas y laboratorios en hojas impresas

Estos fueron los principales procesos analizados para plantear una solución o mejorar en la gestión académica de la institución educativa.

Proceso de Resgistro de Asistencia (Actual)

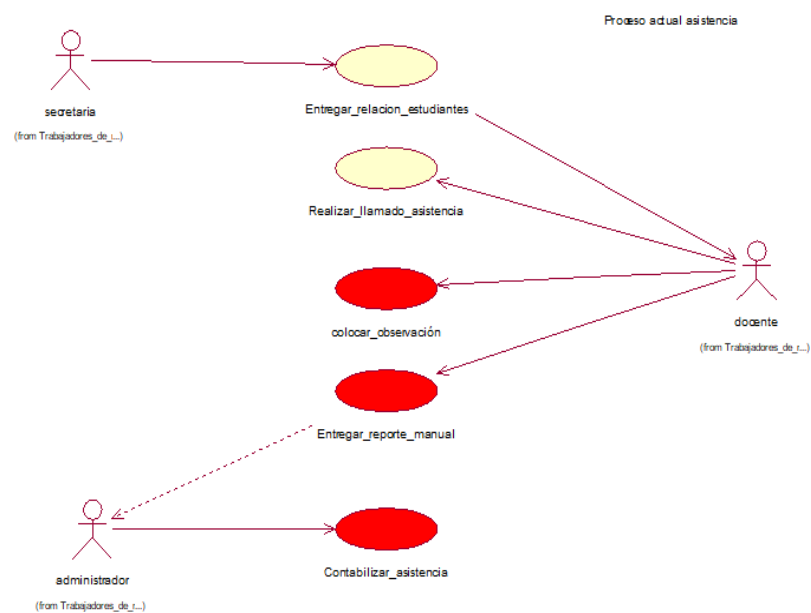


Figura 12. Diagrama de caso de uso Registro de asistencia (actual)

Proceso de Horario (Actual)

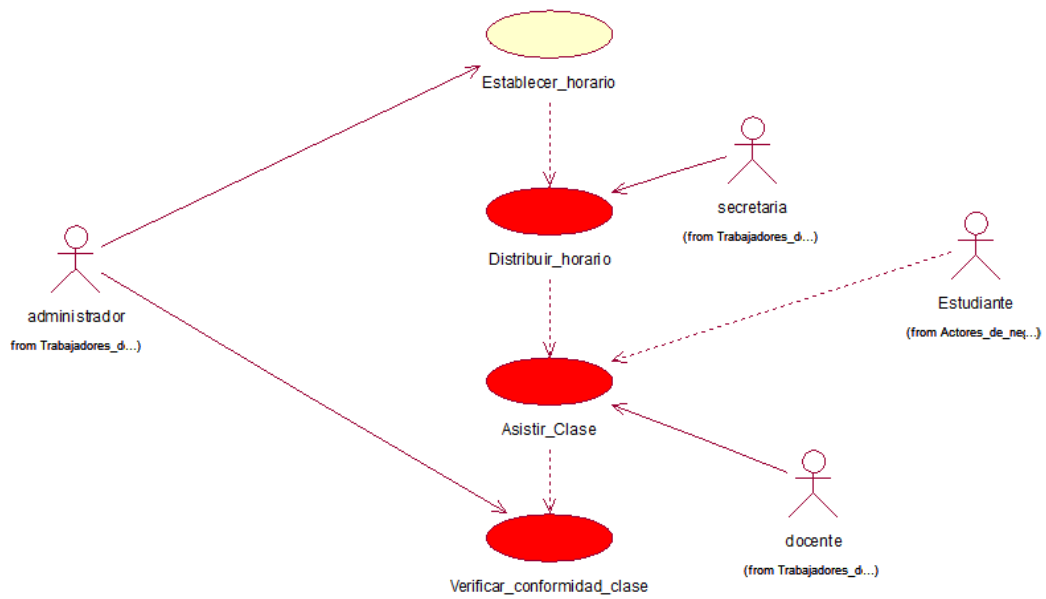


Figura 13. Diagrama de caso de uso Proceso Horario (actual)

Proceso de Registro de Matrícula (Actual)

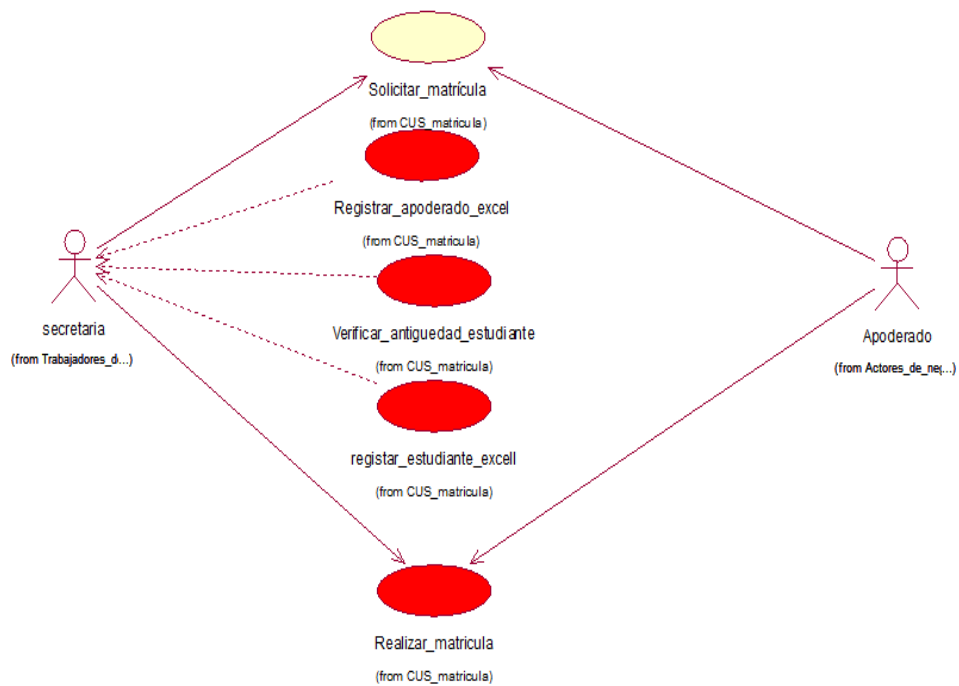


Figura 14. Diagrama de caso de uso Registro de Matrícula (actual)

Proceso de Registro de Notas (actual)

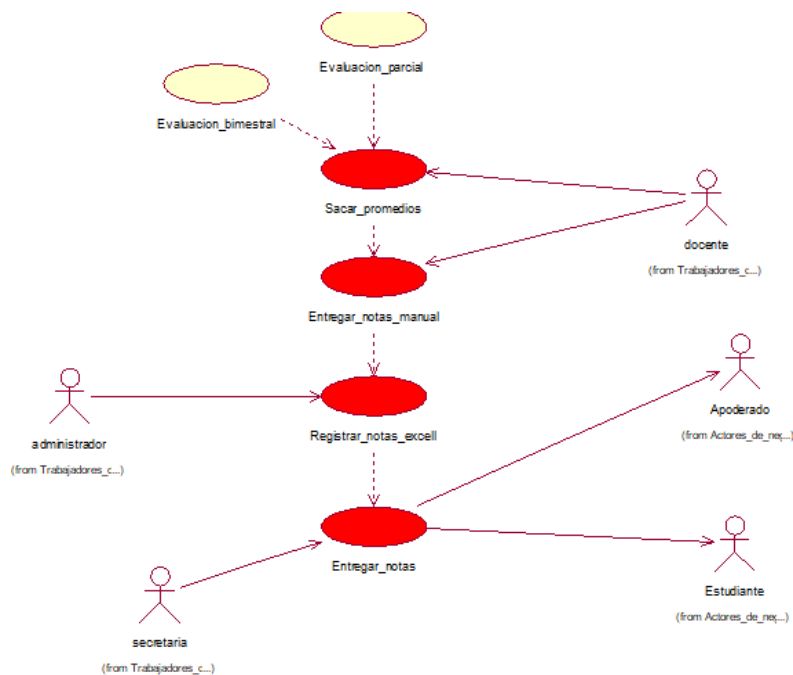


Figura 15. Diagrama de caso de uso Registro de Notas

Análisis de Sistema Propuesto.- En esta etapa de Análisis del sistema propuesto consiste en elaborar los diagramas de caso de uso y el diagrama de clases donde se llega a mostrar las mejoras e innovaciones planteadas que ayudaran a tener un mejor manejo de información en el proceso de gestión académica de la institución educativa.

- Registra las Notas del estudiante
- Registra asistencia del estudiante
- Registra matrícula del estudiante
- Asigna Aulas y laboratorios por grado y sección
- Emite reportes de listas de estudiantes en PDF
- Emite reportes de notas de estudiantes en PDF

Estos son los Procesos que se mejoraron dentro de la gestión académica de la institución educativa Victoria Barcia Boniffatti.

Proceso de Registro de Asistencia (Propuesta)

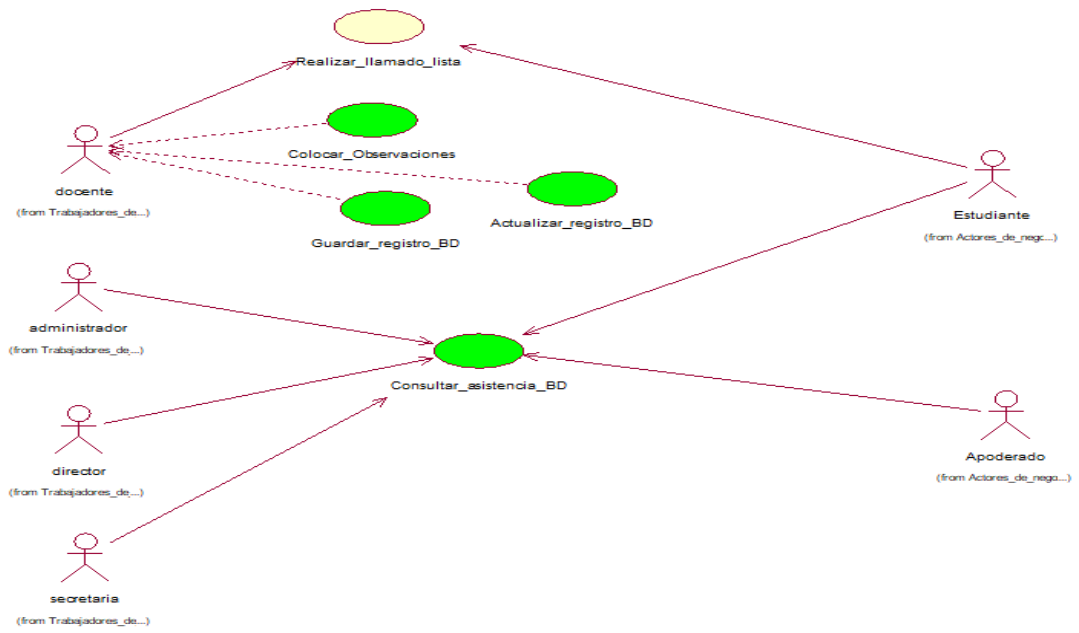


Figura 16. Diagrama de caso de uso Registro de asistencia (Propuesto)

Proceso de Horario (Propuesto)

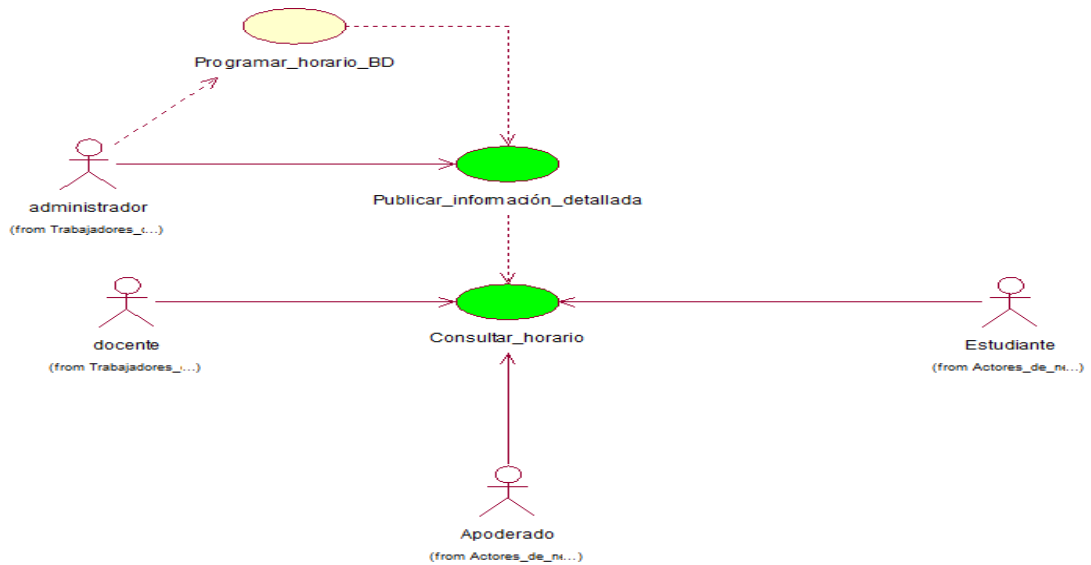


Figura 17. Diagrama de caso de uso Registro de Horario(Propuesta)

Proceso de Registro de Matrícula (Propuesto)

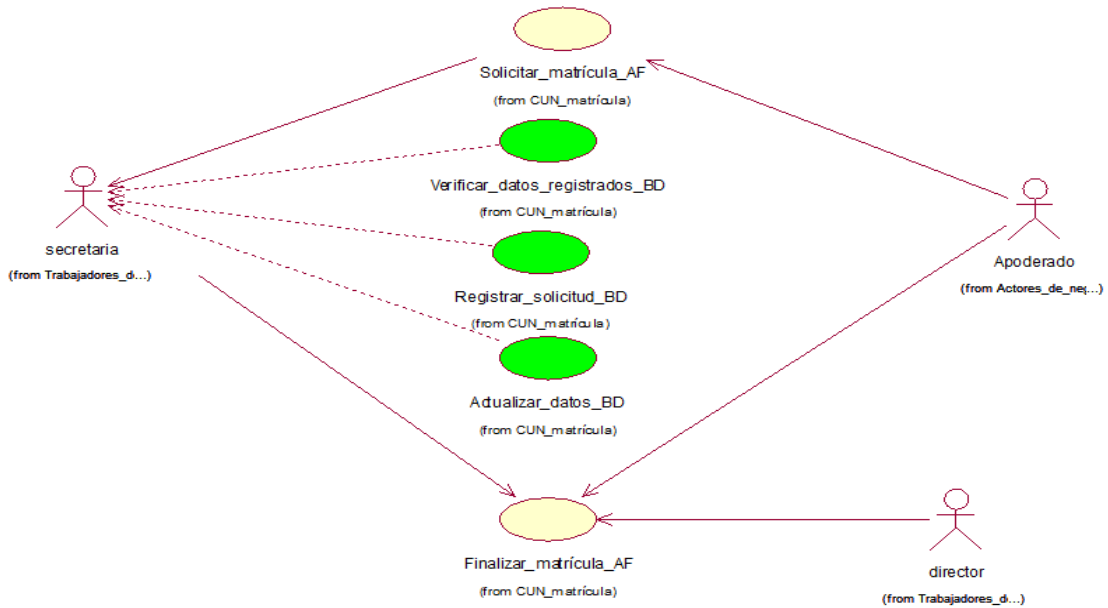


Figura 18. Diagrama de caso de uso Registro de Matricula(Propuesto)

Proceso de Registro de Notas (Propuesto)

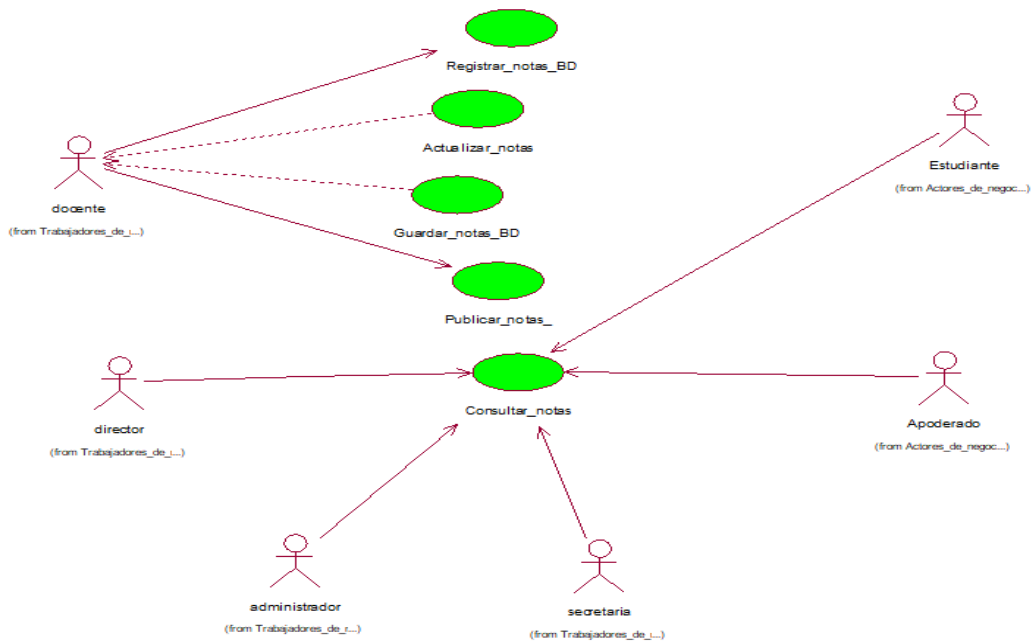


Figura 19. Diagrama de caso de uso Registro de Notas (Propuesto)

Diseño del Sitio web

En esta etapa se diseño el fron-end y el back-end del sistema la cual consistio en la maquetación del menú principal, asi como el diseño de la base de datos

El diseño de la pagina web consiste en integrar su estructura con el uso de la herramientas como el Erwin, PHP, JavaScript, CSS, JQuery, HTML, y usando como administrador de base de datos MySQL y el framework Bootstrap asi como la usabilidad, accesibilidad y organización del sitio web.

Estructura de la página web

En la estructura se muestra la distribución de las diferente opciones de la pagina principal del sitio web es la siguiente:

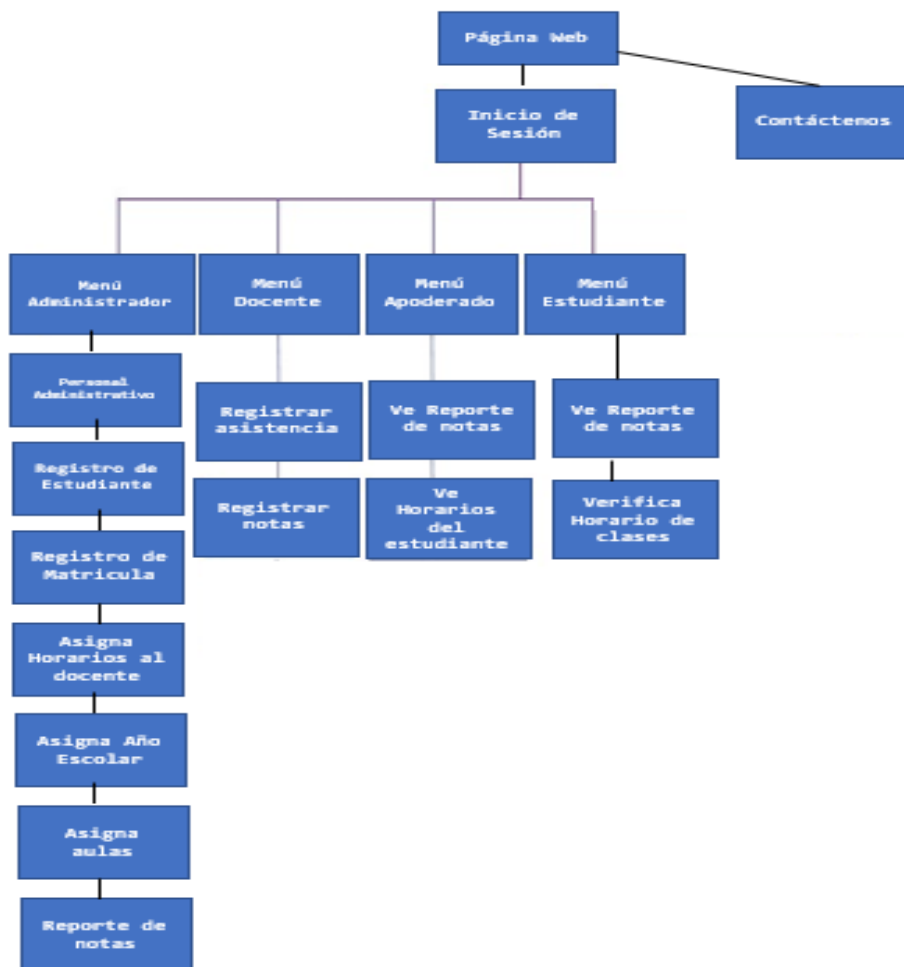


Figura 20. Estructura de la pagina principal (Extranet)

De forma general el sitio web ha sido diseñado en dos partes debidamente definidas de la siguiente manera

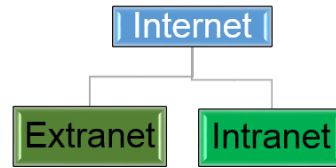


Figura 21. Esquema general del sitio web

Entonces, a continuación se presenta la estructura de la **Extranet** y de la **Intranet**, con las respectivas zonas que conforman la página principal, con las que interactuará el usuario y el administrador respectivamente.

Maquetación de la página principal de la Extranet

ZONA LOGO INSTITUCION	ZONA MENÚ DE SESIÓN
BARRA DE MENU DE NAVEGACIÓN	
ZONA SLIDER DE IMÁGENES	
ZONA NOMBRE DE LA INSTITUCION EDUCATIVA	
ZONA DE MISION, VISION, VALORES	
ZONA DE PROPUESTA ACADEMICA	ZONA DE IMAGEN INSTITUCIONAL
ZONA DE VENTAJAS	
ZONA DE TESTIMONIOS	
ZONA DE FRASE MOTIVACIONAL	
ZONA DE ACTIVIDADES DIARIAS	
LINK CATEGORIAS	CONTACTANOS

Figura 22. Maquetacion de Pagina Principal

Estructura de la página principal de la Intranet

La estructura donde se muestra la distribución de las diferentes opciones de la aplicación web (sitemap.xml).

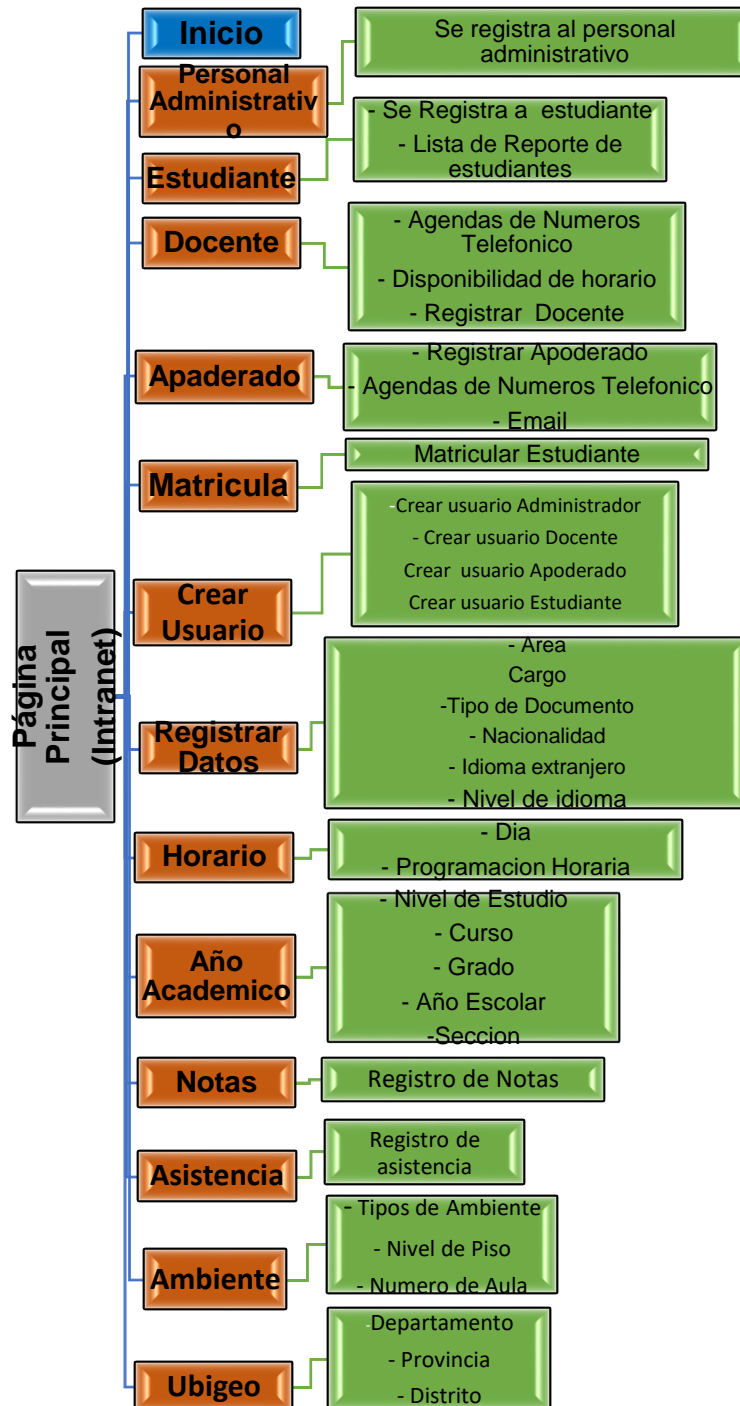


Figura 23. Estructura del Sistemas

Diagrama Entidad – Relación

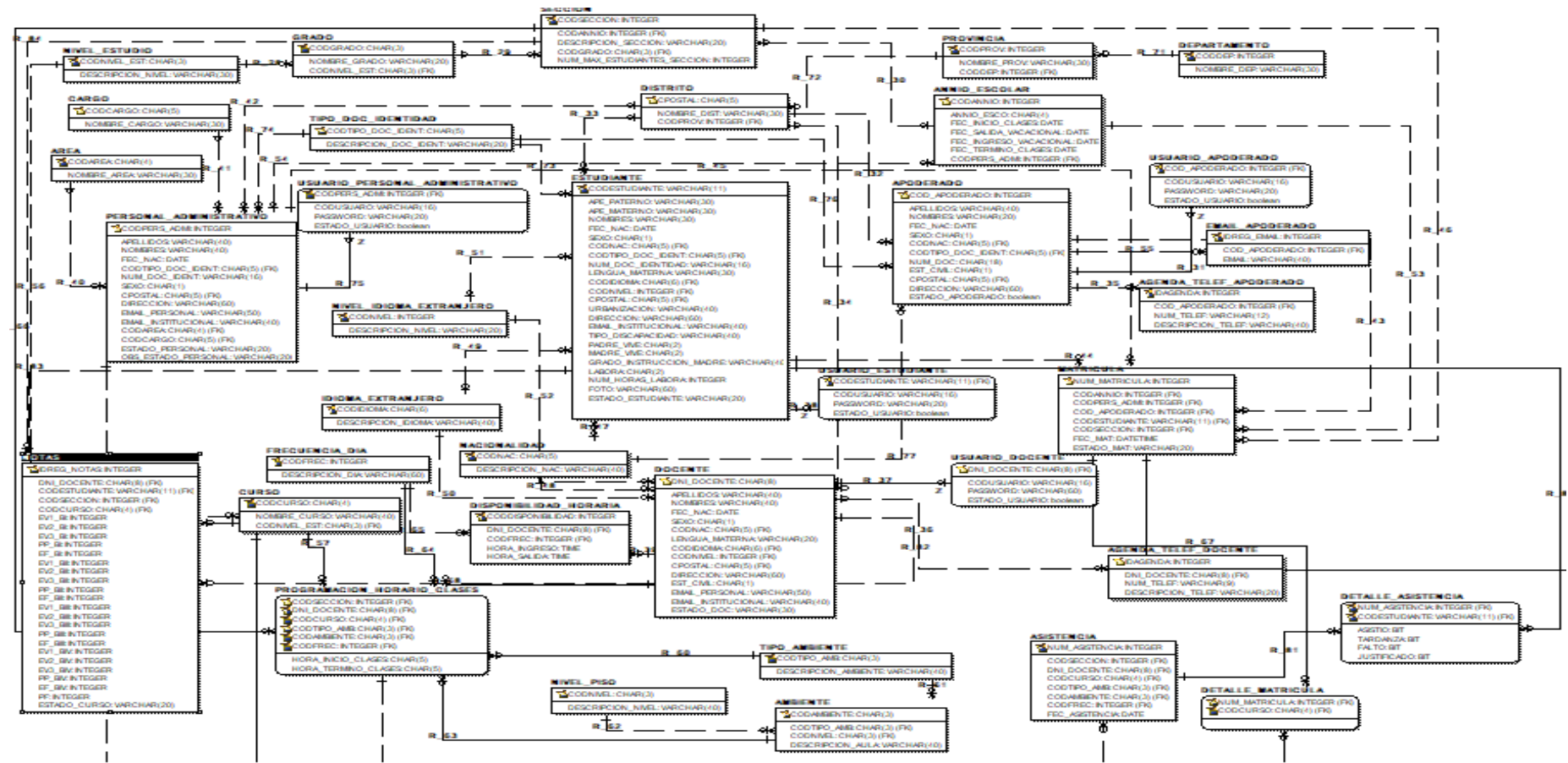


Figura 24. Diagrama Entidad Relación

Script de la Base de Datos de la Aplicación Web Gestion Académica para la Institucion Educativo Victoria Barcia Boniffatti

```
CREATE DATABASE BDGESTION_ACADEMICA_VBB;  
USE BDGESTION_ACADEMICA_VBB;
```

```
CREATE TABLE AGENDA_TELEF_APODERADO  
(  
IDAGENDA      INTEGER AUTO_INCREMENT PRIMARY KEY,  
COD_APODERADO INTEGER NOT NULL,  
NUM_TELEF     VARCHAR(12) NOT NULL,  
DESCRIPCION_TELEF VARCHAR(40) NOT NULL  
);
```

```
CREATE TABLE AGENDA_TELEF_DOCENTE  
(  
IDAGENDA      INTEGER AUTO_INCREMENT PRIMARY KEY,  
DNI_DOCENTE   CHAR(8) NULL,  
NUM_TELEF     VARCHAR(9) NOT NULL,  
DESCRIPCION_TELEF VARCHAR(20) NOT NULL  
);
```

```
CREATE TABLE AMBIENTE  
(  
CODAMBIENTE   CHAR(3) NOT NULL,  
CODTIPO_AMB   CHAR(3) NOT NULL,  
CODNIVEL      CHAR(3) NOT NULL,  
DESCRIPCION_AULA VARCHAR(40) NOT NULL  
);
```

```
ALTER TABLE AMBIENTE  
ADD PRIMARY KEY (CODAMBIENTE);
```

```
CREATE TABLE ANNIO_ESCOLAR  
(  
CODANNIO      INTEGER AUTO_INCREMENT PRIMARY KEY,  
ANNIO_ESCO    CHAR(4) NOT NULL,  
FEC_INICIO_CLASES DATE NOT NULL,  
FEC_SALIDA_VACACIONAL DATE NOT NULL,  
FEC_INGRESO_VACACIONAL DATE NOT NULL,  
FEC_TERMINO_CLASES DATE NOT NULL,  
CODPERS_ADMI  INTEGER NULL  
);
```

```
CREATE TABLE APODERADO  
(  
COD_APODERADO INTEGER AUTO_INCREMENT PRIMARY KEY,  
APELLIDOS     VARCHAR(40) NOT NULL,  
NOMBRES       VARCHAR(20) NOT NULL,  
FEC_NAC       DATE NOT NULL,
```

```

SEXO          CHAR(1) NOT NULL,
CODNAC        CHAR(5) NOT NULL,
CODTIPO_DOC_IDENT CHAR(5) NOT NULL,
NUM_DOC       CHAR(18) NOT NULL,
EST_CIVIL     CHAR(1) NOT NULL,
CPOSTAL       CHAR(5) NOT NULL,
DIRECCION     VARCHAR(60) NOT NULL,
ESTADO_APODERADO VARCHAR(25) NOT NULL
);

```

CREATE TABLE AREA

```

(
CODAREA       CHAR(4) NOT NULL,
NOMBRE_AREA   VARCHAR(30) NOT NULL
);

```

ALTER TABLE AREA

```

ADD PRIMARY KEY (CODAREA);

```

CREATE TABLE ASISTENCIA

```

(
NUM_ASISTENCIA INTEGER AUTO_INCREMENT PRIMARY KEY,
CODSECCION     INTEGER NOT NULL,
DNI_DOCENTE    CHAR(8) NOT NULL,
CODCURSO       CHAR(4) NOT NULL,
CODTIPO_AMB    CHAR(3) NOT NULL,
CODAMBIENTE    CHAR(3) NOT NULL,
CODFREC        INTEGER NOT NULL,
FEC_ASISTENCIA DATE NOT NULL
);

```

CREATE TABLE CARGO

```

(
CODCARGO       CHAR(5) NOT NULL,
NOMBRE_CARGO   VARCHAR(30) NOT NULL
);

```

ALTER TABLE CARGO

```

ADD PRIMARY KEY (CODCARGO);

```

CREATE TABLE CURSO

```

(
CODCURSO       CHAR(4) NOT NULL,
NOMBRE_CURSO   VARCHAR(40) NOT NULL,
CODNIVEL_EST   CHAR(3) NOT NULL
);

```

ALTER TABLE CURSO

ADD PRIMARY KEY (CODCURSO);

CREATE TABLE DEPARTAMENTO

```
(
CODDEP          INTEGER AUTO_INCREMENT PRIMARY KEY,
NOMBRE_DEP      VARCHAR(30) NOT NULL
);
```

CREATE TABLE DETALLE_ASISTENCIA

```
(
NUM_ASISTENCIA  INTEGER NOT NULL,
CODESTUDIANTE  VARCHAR(11) NOT NULL,
ASISTIO         BIT NOT NULL,
TARDANZA       BIT NOT NULL,
FALTO          BIT NOT NULL,
JUSTIFICADO     BIT NOT NULL
);
```

**ALTER TABLE DETALLE_ASISTENCIA
ADD PRIMARY KEY (NUM_ASISTENCIA, CODESTUDIANTE);**

CREATE TABLE DETALLE_MATRÍCULA

```
(
NUM_MATRÍCULA   INTEGER NOT NULL,
CODCURSO        CHAR(4) NOT NULL
);
```

**ALTER TABLE DETALLE_MATRÍCULA
ADD PRIMARY KEY (NUM_MATRÍCULA, CODCURSO);**

CREATE TABLE DISPONIBILIDAD_HORARIA

```
(
CODDISPONIBILIDAD INTEGER AUTO_INCREMENT PRIMARY KEY,
DNI_DOCENTE        CHAR(8) NOT NULL,
CODFREC            INTEGER NOT NULL,
HORA_INGRESO       TIME NOT NULL,
HORA_SALIDA        TIME NOT NULL
);
```

CREATE TABLE DISTRITO

```
(
CPOSTAL          CHAR(5) NOT NULL,
NOMBRE_DIST      VARCHAR(30) NOT NULL,
CODPROV          INTEGER NOT NULL
);
```

```
ALTER TABLE DISTRITO
ADD PRIMARY KEY (CPOSTAL);
```

```
CREATE TABLE DOCENTE
(
DNI_DOCENTE      CHAR(8) NOT NULL,
APELLIDOS        VARCHAR(40) NOT NULL,
NOMBRES          VARCHAR(40) NOT NULL,
FEC_NAC          DATE NOT NULL,
SEXO             CHAR(1) NOT NULL,
CODNAC           CHAR(5) NOT NULL,
LENGUA_MATERNA  VARCHAR(20) NOT NULL,
CODIDIOMA        CHAR(6) NOT NULL,
CODNIVEL         INTEGER NOT NULL,
CPOSTAL          CHAR(5) NULL,
DIRECCION        VARCHAR(60) NOT NULL,
EST_CIVIL        CHAR(1) NOT NULL,
EMAIL_PERSONAL  VARCHAR(50) NOT NULL,
EMAIL_INSTITUCIONAL VARCHAR(40) NOT NULL,
ESTADO_DOC       VARCHAR(30) NOT NULL
);
```

```
ALTER TABLE DOCENTE
ADD PRIMARY KEY (DNI_DOCENTE);
```

```
CREATE TABLE EMAIL_APODERADO
(
IDREG_EMAIL      INTEGER AUTO_INCREMENT PRIMARY KEY,
COD_APODERADO    INTEGER NOT NULL,
EMAIL            VARCHAR(40) NOT NULL
);
```

```
CREATE TABLE ESTUDIANTE
(
CODESTUDIANTE   VARCHAR(11) NOT NULL,
APE_PATERNO     VARCHAR(30) NOT NULL,
APE_MATERNO     VARCHAR(30) NOT NULL,
NOMBRES         VARCHAR(30) NOT NULL,
FEC_NAC         DATE NOT NULL,
SEXO            CHAR(1) NOT NULL,
CODNAC          CHAR(5) NULL,
CODTIPO_DOC_IDENT CHAR(5) NOT NULL,
NUM_DOC_IDENTIDAD VARCHAR(16) NOT NULL,
LENGUA_MATERNA VARCHAR(30) NOT NULL,
CODIDIOMA       CHAR(6) NULL,
CODNIVEL        INTEGER NULL,
CPOSTAL         CHAR(5) NULL,
URBANIZACION    VARCHAR(40) NOT NULL,
DIRECCION       VARCHAR(60) NOT NULL,
EMAIL_INSTITUCIONAL VARCHAR(40) NOT NULL,
TIPO_DISCAPACIDAD VARCHAR(40) NOT NULL,
```



```

PADRE_VIVE      CHAR(2) NOT NULL,
MADRE_VIVE      CHAR(2) NOT NULL,
GRADO_INSTRUCCION_MADRE VARCHAR(40) NOT NULL,
LABORA          CHAR(2) NOT NULL,
NUM_HORAS_LABORA INTEGER NOT NULL,
FOTO            VARCHAR(60) NOT NULL,
ESTADO_ESTUDIANTE VARCHAR(20) NOT NULL
);

```

```

ALTER TABLE ESTUDIANTE
ADD PRIMARY KEY (CODESTUDIANTE);

```

```

CREATE TABLE FRECUENCIA_DIA
(
  CODFREC      INTEGER AUTO_INCREMENT PRIMARY KEY,
  DESCRIPCION_DIA VARCHAR(60) NOT NULL
);

```

```

CREATE TABLE GRADO
(
  CODGRADO      CHAR(3) NOT NULL,
  NOMBRE_GRADO  VARCHAR(20) NOT NULL,
  CODNIVEL_EST  CHAR(3) NOT NULL
);

```

```

ALTER TABLE GRADO
ADD PRIMARY KEY (CODGRADO);

```

```

CREATE TABLE IDIOMA_EXTRANJERO
(
  CODIDIOMA      CHAR(6) NOT NULL,
  DESCRIPCION_IDIOMA VARCHAR(40) NOT NULL
);

```

```

ALTER TABLE IDIOMA_EXTRANJERO
ADD PRIMARY KEY (CODIDIOMA);

```

```

CREATE TABLE MATRÍCULA
(
  NUM_MATRÍCULA  INTEGER AUTO_INCREMENT PRIMARY KEY,
  CODANNIO       INTEGER NOT NULL,
  CODPERS_ADMI   INTEGER NOT NULL,
  COD_APODERADO  INTEGER NOT NULL,
  CODESTUDIANTE VARCHAR(11) NOT NULL,
  CODESECCION    INTEGER NOT NULL,
  FEC_MAT        DATETIME NOT NULL,

```

```
ESTADO_MAT      VARCHAR(20) NOT NULL
);
```

```
CREATE TABLE NACIONALIDAD
(
CODNAC          CHAR(5) NOT NULL,
DESCRIPCION_NAC VARCHAR(40) NOT NULL
);
```

```
ALTER TABLE NACIONALIDAD
ADD PRIMARY KEY (CODNAC);
```

```
CREATE TABLE NIVEL_ESTUDIO
(
CODNIVEL_EST    CHAR(3) NOT NULL,
DESCRIPCION_NIVEL VARCHAR(30) NOT NULL
);
```

```
ALTER TABLE NIVEL_ESTUDIO
ADD PRIMARY KEY (CODNIVEL_EST);
```

```
CREATE TABLE NIVEL_IDIOMA_EXTRANJERO
(
CODNIVEL        INTEGER AUTO_INCREMENT PRIMARY KEY,
DESCRIPCION_NIVEL VARCHAR(20) NOT NULL
);
```

```
CREATE TABLE NIVEL_PISO
(
CODNIVEL        CHAR(3) NOT NULL,
DESCRIPCION_NIVEL VARCHAR(40) NOT NULL
);
```

```
ALTER TABLE NIVEL_PISO
ADD PRIMARY KEY (CODNIVEL);
```

```
CREATE TABLE NOTAS
(
IDREG_NOTAS     INTEGER NOT NULL,
DNI_DOCENTE     CHAR(8) NOT NULL,
CODESTUDIANTE   VARCHAR(11) NOT NULL,
CODSECCION      INTEGER NOT NULL,
CODCURSO        CHAR(4) NOT NULL,
EV1_BI          INTEGER NOT NULL,
EV2_BI          INTEGER NOT NULL,
EV3_BI          INTEGER NOT NULL,
```

```

PP_BI          INTEGER NOT NULL,
EF_BI          INTEGER NOT NULL,
EV1_BII        INTEGER NOT NULL,
EV2_BII        INTEGER NOT NULL,
EV3_BII        INTEGER NOT NULL,
PP_BII        INTEGER NOT NULL,
EF_BII        INTEGER NOT NULL,
EV1_BIII       INTEGER NOT NULL,
EV2_BIII       INTEGER NOT NULL,
EV3_BIII       INTEGER NOT NULL,
PP_BIII        INTEGER NOT NULL,
EF_BIII        INTEGER NOT NULL,
EV1_BIV        INTEGER NOT NULL,
EV2_BIV        INTEGER NOT NULL,
EV3_BIV        INTEGER NOT NULL,
PP_BIV        INTEGER NOT NULL,
EF_BIV        INTEGER NOT NULL,
PF             INTEGER NOT NULL,
ESTADO_CURSO   VARCHAR(20) NOT NULL
);

```

```

ALTER TABLE NOTAS
ADD PRIMARY KEY (IDREG_NOTAS);

```

```

CREATE TABLE PAGO_PENSION
(
NUM_MATRÍCULA  INTEGER NOT NULL,
NUM_CUOTA      INTEGER NOT NULL,
FEC_PAGO       DATE NOT NULL,
VALOR_PENSION  NUMERIC(10,2) NOT NULL,
FEC_PAGO_EFECTUADO DATETIME NOT NULL,
VALOR_CANCELADO NUMERIC(10,2) NOT NULL,
INTERES        NUMERIC(10,2) NOT NULL,
CODPERS_ADMI   INTEGER NOT NULL,
ESTADO_CUOTA_PENSION VARCHAR(20) NOT NULL
);

```

```

ALTER TABLE PAGO_PENSION
ADD PRIMARY KEY (NUM_MATRÍCULA,NUM_CUOTA);

```

```

CREATE TABLE PERSONAL_ADMINISTRATIVO
(
CODPERS_ADMI   INTEGER AUTO_INCREMENT PRIMARY KEY,
APELLIDOS      VARCHAR(40) NOT NULL,
NOMBRES        VARCHAR(40) NOT NULL,
FEC_NAC        DATE NOT NULL,
CODTIPO_DOC_IDENT CHAR(5) NOT NULL,
NUM_DOC_IDENT  VARCHAR(16) NOT NULL,
SEXO           CHAR(1) NOT NULL,
CPOSTAL        CHAR(5) NULL,
DIRECCION      VARCHAR(60) NOT NULL,
EMAIL_PERSONAL VARCHAR(50) NOT NULL,

```

```

EMAIL_INSTITUCIONAL VARCHAR(40) NOT NULL,
CODAREA          CHAR(4) NULL,
CODCARGO         CHAR(5) NULL,
ESTADO_PERSONAL  VARCHAR(20) NOT NULL,
OBS_ESTADO_PERSONAL VARCHAR(200) NOT NULL
);

```

```

CREATE TABLE PROGRAMACION_HORARIO_CLASES
(
CODSECCION      INTEGER NOT NULL,
DNI_DOCENTE     CHAR(8) NOT NULL,
CODCURSO        CHAR(4) NOT NULL,
CODTIPO_AMB     CHAR(3) NOT NULL,
CODAMBIENTE     CHAR(3) NOT NULL,
CODFREC         INTEGER NOT NULL,
HORA_INICIO_CLASES CHAR(5) NOT NULL,
HORA_TERMINO_CLASES CHAR(5) NOT NULL
);

```

```

ALTER TABLE PROGRAMACION_HORARIO_CLASES
ADD PRIMARY
KEY(CODSECCION,DNI_DOCENTE,CODCURSO,CODTIPO_AMB,CODAMBIENTE,CODFREC)

```

```

CREATE TABLE PROVINCIA
(
CODPROV        INTEGER AUTO_INCREMENT PRIMARY KEY,
NOMBRE_PROV    VARCHAR(30) NOT NULL,
CODDEP         INTEGER NOT NULL
);

```

```

CREATE TABLE SECCION
(
CODSECCION     INTEGER AUTO_INCREMENT PRIMARY KEY,
CODANNIO       INTEGER NULL,
DESCRIPCION_SECCION VARCHAR(20) NOT NULL,
CODGRADO       CHAR(3) NULL,
NUM_MAX_ESTUDIANTES_SECCION INTEGER NOT NULL
);

```

```

CREATE TABLE TIPO_AMBIENTE
(
CODTIPO_AMB    CHAR(3) NOT NULL,
DESCRIPCION_AMBIENTE VARCHAR(40) NOT NULL
);

```

```

ALTER TABLE TIPO_AMBIENTE
ADD PRIMARY KEY (CODTIPO_AMB);

```

```
CREATE TABLE TIPO_DOC_IDENTIDAD  
(  
  CODTIPO_DOC_IDENT CHAR(5) NOT NULL,  
  DESCRIPCION_DOC_IDENT VARCHAR(20) NOT NULL  
);
```

```
ALTER TABLE TIPO_DOC_IDENTIDAD  
ADD PRIMARY KEY (CODTIPO_DOC_IDENT);
```

```
CREATE TABLE USUARIO_APODERADO  
(  
  COD_APODERADO INTEGER NOT NULL,  
  CODUSUARIO VARCHAR(16) NOT NULL,  
  PASSWORD VARCHAR(20) NOT NULL,  
  ESTADO_USUARIO boolean NOT NULL  
);
```

```
ALTER TABLE USUARIO_APODERADO  
ADD PRIMARY KEY (COD_APODERADO);
```

```
CREATE TABLE USUARIO_DOCENTE  
(  
  DNI_DOCENTE CHAR(8) NOT NULL,  
  CODUSUARIO VARCHAR(16) NOT NULL,  
  PASSWORD VARCHAR(60) NOT NULL,  
  ESTADO_USUARIO boolean NOT NULL  
);
```

```
ALTER TABLE USUARIO_DOCENTE  
ADD PRIMARY KEY (DNI_DOCENTE);
```

```
CREATE TABLE USUARIO_ESTUDIANTE  
(  
  CODESTUDIANTE VARCHAR(11) NOT NULL,  
  CODUSUARIO VARCHAR(16) NOT NULL,  
  PASSWORD VARCHAR(20) NOT NULL,  
  ESTADO_USUARIO boolean NOT NULL  
);
```

```
ALTER TABLE USUARIO_ESTUDIANTE  
ADD PRIMARY KEY (CODESTUDIANTE);
```

```
CREATE TABLE USUARIO_PERSONAL_ADMINISTRATIVO  
(
```

```
CODPERS_ADMI    INTEGER NOT NULL,  
CODUSUARIO     VARCHAR(16) NOT NULL,  
PASSWORD       VARCHAR(20) NOT NULL,  
ESTADO_USUARIO boolean NOT NULL  
);
```

```
ALTER TABLE USUARIO_PERSONAL_ADMINISTRATIVO  
ADD PRIMARY KEY (CODPERS_ADMI);
```

```
ALTER TABLE AGENDA_TELEF_APODERADO  
ADD FOREIGN KEY R_35 (COD_APODERADO) REFERENCES APODERADO  
(COD_APODERADO);
```

```
ALTER TABLE AGENDA_TELEF_DOCENTE  
ADD FOREIGN KEY R_36 (DNI_DOCENTE) REFERENCES DOCENTE (DNI_DOCENTE);
```

```
ALTER TABLE AMBIENTE  
ADD FOREIGN KEY R_61 (CODTIPO_AMB) REFERENCES TIPO_AMBIENTE  
(CODTIPO_AMB);
```

```
ALTER TABLE AMBIENTE  
ADD FOREIGN KEY R_62 (CODNIVEL) REFERENCES NIVEL_PISO (CODNIVEL);
```

```
ALTER TABLE ANNIO_ESCOLAR  
ADD FOREIGN KEY R_54 (CODPERS_ADMI) REFERENCES PERSONAL_ADMINISTRATIVO  
(CODPERS_ADMI);
```

```
ALTER TABLE APODERADO  
ADD FOREIGN KEY R_32 (CPOSTAL) REFERENCES DISTRITO (CPOSTAL);
```

```
ALTER TABLE APODERADO  
ADD FOREIGN KEY R_76 (CODTIPO_DOC_IDENT) REFERENCES TIPO_DOC_IDENTIDAD  
(CODTIPO_DOC_IDENT);
```

```
ALTER TABLE APODERADO  
ADD FOREIGN KEY R_77 (CODNAC) REFERENCES NACIONALIDAD (CODNAC);
```

```
ALTER TABLE ASISTENCIA
```

ADD FOREIGN KEY R_70 (CODSECCION, DNI_DOCENTE, CODCURSO, CODTIPO_AMB, CODAMBIENTE, CODFREC) **REFERENCES** PROGRAMAION_HORARIO_CLASES (CODSECCION, DNI_DOCENTE, CODCURSO, CODTIPOAMB, CODAMBIENTE, CODFREC);

ALTER TABLE CURSO
ADD FOREIGN KEY R_56 (CODNIVEL_EST) **REFERENCES** NIVEL_ESTUDIO(CODNIVEL_EST);

ALTER TABLE DETALLE_ASISTENCIA
ADD FOREIGN KEY R_80 (CODESTUDIANTE) **REFERENCES** ESTUDIANTE(CODESTUDIANTE);

ALTER TABLE DETALLE_ASISTENCIA
ADD FOREIGN KEY R_81 (NUM_ASISTENCIA) **REFERENCES** ASISTENCIA(NUM_ASISTENCIA);

ALTER TABLE DETALLE_MATRÍCULA
ADD FOREIGN KEY R_67 (NUM_MATRÍCULA) **REFERENCES** MATRÍCULA(NUM_MATRÍCULA);

ALTER TABLE DETALLE_MATRÍCULA
ADD FOREIGN KEY R_68 (CODCURSO) **REFERENCES** CURSO (CODCURSO);

ALTER TABLE DISPONIBILIDAD_HORARIA
ADD FOREIGN KEY R_39 (DNI_DOCENTE) **REFERENCES** DOCENTE(DNI_DOCENTE);

ALTER TABLE DISPONIBILIDAD_HORARIA
ADD FOREIGN KEY R_65 (CODFREC) **REFERENCES** FRECUENCIA_DIA (CODFREC);

ALTER TABLE DISTRITO
ADD FOREIGN KEY R_72 (CODPROV) **REFERENCES** PROVINCIA (CODPROV);

ALTER TABLE DOCENTE
ADD FOREIGN KEY R_34 (CPOSTAL) **REFERENCES** DISTRITO (CPOSTAL);

ALTER TABLE DOCENTE
ADD FOREIGN KEY R_48 (CODNAC) **REFERENCES** NACIONALIDAD (CODNAC);

ALTER TABLE DOCENTE
ADD FOREIGN KEY R_50 (CODIDIOMA) **REFERENCES** IDIOMA_EXTRANJERO
(CODIDIOMA);

ALTER TABLE DOCENTE
ADD FOREIGN KEY R_52 (CODNIVEL) **REFERENCES**
NIVEL_IDIOMA_EXTRANJERO(CODNIVEL);

ALTER TABLE EMAIL_APODERADO
ADD FOREIGN KEY R_55 (COD_APODERADO) **REFERENCES**
APODERADO(COD_APODERADO);

ALTER TABLE ESTUDIANTE
ADD FOREIGN KEY R_33 (CPOSTAL) **REFERENCES** DISTRITO (CPOSTAL);

ALTER TABLE ESTUDIANTE
ADD FOREIGN KEY R_47 (CODNAC) **REFERENCES** NACIONALIDAD (CODNAC);

ALTER TABLE ESTUDIANTE
ADD FOREIGN KEY R_49 (CODIDIOMA) **REFERENCES**
IDIOMA_EXTRANJERO(CODIDIOMA);

ALTER TABLE ESTUDIANTE
ADD FOREIGN KEY R_51 (CODNIVEL) **REFERENCES**
NIVEL_IDIOMA_EXTRANJERO(CODNIVEL);

ALTER TABLE ESTUDIANTE
ADD FOREIGN KEY R_73 (CODTIPO_DOC_IDENT) **REFERENCES** TIPO_DOC_IDENTIDAD
(CODTIPO_DOC_IDENT);

ALTER TABLE GRADO
ADD FOREIGN KEY R_28 (CODNIVEL_EST) **REFERENCES**
NIVEL_ESTUDIO(CODNIVEL_EST);

ALTER TABLE MATRÍCULA
ADD FOREIGN KEY R_43 (COD_APODERADO) **REFERENCES**
APODERADO(COD_APODERADO);

ALTER TABLE MATRÍCULA

ADD FOREIGN KEY R_44 (CODESTUDIANTE) REFERENCES ESTUDIANTE(CODESTUDIANTE);

**ALTER TABLE MATRÍCULA
ADD FOREIGN KEY R_45 (CODPERS_ADMI) REFERENCES PERSONAL_ADMINISTRATIVO (CODPERS_ADMI);**

**ALTER TABLE MATRÍCULA
ADD FOREIGN KEY R_46 (CODSECCION) REFERENCES SECCION (CODSECCION);**

**ALTER TABLE MATRÍCULA
ADD FOREIGN KEY R_53 (CODANNIO) REFERENCES ANNIO_ESCOLAR (CODANNIO);**

**ALTER TABLE NOTAS
ADD FOREIGN KEY R_82 (DNI_DOCENTE) REFERENCES DOCENTE (DNI_DOCENTE);**

**ALTER TABLE NOTAS
ADD FOREIGN KEY R_83 (CODESTUDIANTE) REFERENCES ESTUDIANTE (CODESTUDIANTE);**

**ALTER TABLE NOTAS
ADD FOREIGN KEY R_84 (CODSECCION) REFERENCES SECCION (CODSECCION);**

**ALTER TABLE NOTAS
ADD FOREIGN KEY R_85 (CODCURSO) REFERENCES CURSO (CODCURSO);**

**ALTER TABLE PAGO_PENSION
ADD FOREIGN KEY R_78 (NUM_MATRÍCULA) REFERENCES MATRÍCULA (NUM_MATRÍCULA);**

**ALTER TABLE PAGO_PENSION
ADD FOREIGN KEY R_79 (CODPERS_ADMI) REFERENCES PERSONAL_ADMINISTRATIVO (CODPERS_ADMI);**

**ALTER TABLE PERSONAL_ADMINISTRATIVO
ADD FOREIGN KEY R_40 (CODAREA) REFERENCES AREA (CODAREA);**

ALTER TABLE PERSONAL_ADMINISTRATIVO

ADD FOREIGN KEY R_41 (CODCARGO) REFERENCES CARGO (CODCARGO);

**ALTER TABLE PERSONAL_ADMINISTRATIVO
ADD FOREIGN KEY R_42 (CPOSTAL) REFERENCES DISTRITO (CPOSTAL);**

**ALTER TABLE PERSONAL_ADMINISTRATIVO
ADD FOREIGN KEY R_74 (CODTIPO_DOC_IDENT) REFERENCES TIPO_DOC_IDENTIDAD
(CODTIPO_DOC_IDENT);**

**ALTER TABLE PROGRAMACION_HORARIO_CLASES
ADD FOREIGN KEY R_57 (CODCURSO) REFERENCES CURSO (CODCURSO);**

**ALTER TABLE PROGRAMACION_HORARIO_CLASES
ADD FOREIGN KEY R_58 (DNI_DOCENTE) REFERENCES DOCENTE (DNI_DOCENTE);**

**ALTER TABLE PROGRAMACION_HORARIO_CLASES
ADD FOREIGN KEY R_60 (CODTIPO_AMB) REFERENCES TIPO_AMBIENTE
(CODTIPO_AMB);**

**ALTER TABLE PROGRAMACION_HORARIO_CLASES
ADD FOREIGN KEY R_63 (CODAMBIENTE) REFERENCES AMBIENTE (CODAMBIENTE);**

**ALTER TABLE PROGRAMACION_HORARIO_CLASES
ADD FOREIGN KEY R_64 (CODFREC) REFERENCES FRECUENCIA_DIA (CODFREC);**

**ALTER TABLE PROGRAMACION_HORARIO_CLASES
ADD FOREIGN KEY R_66 (CODSECCION) REFERENCES SECCION (CODSECCION);**

**ALTER TABLE PROVINCIA
ADD FOREIGN KEY R_71 (CODDEP) REFERENCES DEPARTAMENTO (CODDEP);**

**ALTER TABLE SECCION
ADD FOREIGN KEY R_29 (CODGRADO) REFERENCES GRADO (CODGRADO);**

**ALTER TABLE SECCION
ADD FOREIGN KEY R_30 (CODANNIO) REFERENCES ANNIO_ESCOLAR (CODANNIO);**

```

ALTER TABLE USUARIO_APODERADO
ADD FOREIGN KEY R_31 (COD_APODERADO) REFERENCES APODERADO
(COD_APODERADO);

```

```

ALTER TABLE USUARIO_DOCENTE
ADD FOREIGN KEY R_37 (DNI_DOCENTE) REFERENCES DOCENTE (DNI_DOCENTE);

```

```

ALTER TABLE USUARIO_ESTUDIANTE
ADD FOREIGN KEY R_38 (CODESTUDIANTE) REFERENCES ESTUDIANTE
(CODESTUDIANTE);

```

```

ALTER TABLE USUARIO_PERSONAL_ADMINISTRATIVO
ADD FOREIGN KEY R_75 (CODPERS_ADMI) REFERENCES
PERSONAL_ADMINISTRATIVO (CODPERS_ADMI);

```

```

DROP PROCEDURE IF EXISTS SP_BUSDEL_AREA;

```

```

DELIMITER $$

```

```

CREATE PROCEDURE SP_BUSDEL_AREA

```

```

(
BUS VARCHAR(80),
TIPO CHAR(1)
)

```

```

BEGIN
IF TIPO='B' THEN
SELECT * FROM AREA WHERE
CODAREA LIKE CONCAT('%',BUS,'%') OR
NOMBRE_AREA LIKE CONCAT('%',BUS,'%');
ELSEIF TIPO='D' THEN
DELETE FROM AREA WHERE
CODAREA=BUS;
ELSEIF TIPO='E' THEN
SELECT * FROM AREA WHERE
CODAREA=BUS;
END IF;
END$$
DELIMITER $$

```

```

CALL SP_BUSDEL_AREA('B')

```

```

DROP PROCEDURE IF EXISTS SP_ADDUP_AREA;

```

```

DELIMITER $$

```

```

CREATE PROCEDURE SP_ADDUP_AREA(CODA CHAR(4),NOMA VARCHAR(80), TIPO
CHAR(1))

```

```

BEGIN
IF TIPO='G' THEN
INSERT INTO AREA VALUES(CODA,NOMA);
ELSEIF TIPO='A' THEN
UPDATE AREA SET NOMBRE_AREA=NOMA WHERE CODAREA=CODA;
END IF;
END$$

```

DELIMITER \$\$

CALL SP_ADDUP_AREA('Psicología','G');

DROP TRIGGER IF EXISTS GENERAR_CODIGO_AREA;

DELIMITER \$\$

CREATE TRIGGER GENERAR_CODIGO_AREA **BEFORE INSERT ON AREA FOR EACH ROW**

BEGIN

DECLARE NEXT_CODE INT;

SET NEXT_CODE=(**SELECT IFNULL**(**MAX**(**CONVERT**(**SUBSTRING**(CODAREA,2),**SIGNED INTEGER**)),0) **FROM** AREA)+1;

SET NEW.CODAREA=**CONCAT**('A',**LPAD**(NEXT_CODE,3,'0'));

END\$\$

DELIMITER \$\$

DROP PROCEDURE IF EXISTS SP_BUSDEL_CARGO;

DELIMITER \$\$

CREATE PROCEDURE SP_BUSDEL_CARGO

(
BUS **VARCHAR**(80),
TIPO **CHAR**(1)

)

BEGIN

IF TIPO='B' **THEN**

SELECT * FROM CARGO **WHERE**

CODCARGO **LIKE** **CONCAT**('%',BUS,'%') **OR**

NOMBRE_CARGO **LIKE** **CONCAT**('%',BUS,'%');

ELSEIF TIPO='D' **THEN**

DELETE FROM CARGO **WHERE**

CODCARGO=BUS;

ELSEIF TIPO='E' **THEN**

SELECT * FROM CARGO **WHERE**

CODCARGO=BUS;

END IF;

END\$\$

DELIMITER \$\$

CALL SP_BUSDEL_CARGO('B')

DROP PROCEDURE IF EXISTS SP_ADDUP_CARGO;

DELIMITER \$\$

CREATE PROCEDURE SP_ADDUP_CARGO(CODC **CHAR**(5),NOMC **VARCHAR**(40), TIPO **CHAR**(1))

BEGIN

IF TIPO='G' **THEN**

INSERT INTO CARGO **VALUES**(CODC,NOMC);

ELSEIF TIPO='A' **THEN**

UPDATE CARGO **SET** NOMBRE_CARGO=NOMC **WHERE** CODCARGO=CODC;

END IF;

END\$\$

DELIMITER \$\$

CALL SP_ADDUP_CARGO('C003','Docente','G');

DROP TRIGGER IF EXISTS GENERAR_CODIGO_CARGO;

DELIMITER \$\$

CREATE TRIGGER GENERAR_CODIGO_CARGO **BEFORE INSERT ON CARGO FOR EACH ROW**

BEGIN

```

DECLARE NEXT_CODE INT;
SET NEXT_CODE=(SELECT
IFNULL(MAX(CONVERT(SUBSTRING(CODCARGO,2),SIGNED INTEGER)),0) FROM
CARGO)+1;
SET NEW.CODCARGO=CONCAT('C',LPAD(NEXT_CODE,4,'0'));
END$$
DELIMITER $$

```

```

DROP PROCEDURE IF EXISTS SP_BUSDEL_TIPO_DOC_IDENTIDAD;
DELIMITER $$
CREATE PROCEDURE SP_BUSDEL_TIPO_DOC_IDENTIDAD
(
BUS VARCHAR(80),
TIPO CHAR(1)
)
BEGIN
IF TIPO='B' THEN
SELECT * FROM TIPO_DOC_IDENTIDAD WHERE
CODTIPO_DOC_IDENT LIKE CONCAT('%',BUS,'%') OR
DESCRIPCION_DOC_IDENT LIKE CONCAT('%',BUS,'%');
ELSEIF TIPO='D' THEN
DELETE FROM TIPO_DOC_IDENTIDAD WHERE
CODTIPO_DOC_IDENT=BUS;
ELSEIF TIPO='E' THEN
SELECT * FROM TIPO_DOC_IDENTIDAD WHERE
CODTIPO_DOC_IDENT=BUS;
ELSEIF TIPO='L' THEN
SELECT * FROM TIPO_DOC_IDENTIDAD WHERE
CODTIPO_DOC_IDENT=BUS OR DESCRIPCION_DOC_IDENT LIKE CONCAT('%',BUS,'%');
END IF;
END$$
DELIMITER $$

```

```

CALL SP_BUSDEL_TIPO_DOC_IDENTIDAD(",'B')

```

```

DROP PROCEDURE IF EXISTS SP_ADDUP_TIPO_DOC_IDENTIDAD;
DELIMITER $$
CREATE PROCEDURE SP_ADDUP_TIPO_DOC_IDENTIDAD(CODTDI CHAR(5),NOMTDI
VARCHAR(80), TIPO CHAR(1))
BEGIN
IF TIPO='G' THEN
INSERT INTO TIPO_DOC_IDENTIDAD VALUES(CODTDI,NOMTDI);
ELSEIF TIPO='A' THEN
UPDATE TIPO_DOC_IDENTIDAD SET DESCRIPCION_DOC_IDENT=NOMTDI WHERE
CODTIPO_DOC_IDENT=CODTDI;
END IF;
END$$
DELIMITER $$

```

```

CALL SP_ADDUP_TIPO_DOC_IDENTIDAD("','G');

```

```

DROP TRIGGER IF EXISTS GENERAR_CODIGO_TIPO_DOC_IDENTIDAD;
DELIMITER $$
CREATE TRIGGER GENERAR_CODIGO_TIPO_DOC_IDENTIDAD BEFORE INSERT ON
TIPO_DOC_IDENTIDAD FOR EACH ROW
BEGIN
DECLARE NEXT_CODE INT;

```

```

SET NEXT_CODE=(SELECT
IFNULL(MAX(CONVERT(SUBSTRING(CODTIPO_DOC_IDENT,4),SIGNED INTEGER)),0)
FROM TIPO_DOC_IDENTIDAD)+1;
SET NEW.CODTIPO_DOC_IDENT=CONCAT('TDI',LPAD(NEXT_CODE,2,'0'));
END$$
DELIMITER $$

```

```

DROP PROCEDURE IF EXISTS SP_BUSDEL_DEPARTAMENTO;
DELIMITER $$

```

```

CREATE PROCEDURE SP_BUSDEL_DEPARTAMENTO

```

```

(
BUS VARCHAR(80),
TIPO CHAR(1)
)

```

```

BEGIN

```

```

IF TIPO='B' THEN

```

```

SELECT * FROM DEPARTAMENTO WHERE

```

```

CODDEP LIKE CONCAT('%',BUS,'%') OR

```

```

NOMBRE_DEP LIKE CONCAT('%',BUS,'%');

```

```

ELSEIF TIPO='D' THEN

```

```

DELETE FROM DEPARTAMENTO WHERE

```

```

CODDEP=BUS;

```

```

ELSEIF TIPO='E' THEN

```

```

SELECT * FROM DEPARTAMENTO WHERE

```

```

CODDEP=BUS;

```

```

END IF;

```

```

END$$

```

```

DELIMITER $$

```

```

CALL SP_BUSDEL_DEPARTAMENTO('','B')

```

```

DROP PROCEDURE IF EXISTS SP_ADDUP_DEPARTAMENTO;

```

```

DELIMITER $$

```

```

CREATE PROCEDURE SP_ADDUP_DEPARTAMENTO(CODDP INT,NOMDP VARCHAR(40),

```

```

TIPO CHAR(1))

```

```

BEGIN

```

```

IF TIPO='G' THEN

```

```

INSERT INTO DEPARTAMENTO VALUES(NULL,NOMDP);

```

```

ELSEIF TIPO='A' THEN

```

```

UPDATE DEPARTAMENTO SET NOMBRE_DEP=NOMDP WHERE CODDEP=CODDP;

```

```

END IF;

```

```

END$$

```

```

DELIMITER $$

```

```

CALL SP_ADDUP_DEPARTAMENTO(NULL,'Docente','G');

```

```

SELECT * FROM DEPARTAMENTO

```

```

DROP PROCEDURE IF EXISTS SP_BUSDEL_PROVINCIA;

```

```

DELIMITER $$

```

```

CREATE PROCEDURE SP_BUSDEL_PROVINCIA

```

```

(
BUS VARCHAR(80),
TIPO CHAR(1)
)

```

```

)

```

```

BEGIN

```

```

IF TIPO='B' THEN

```

```

SELECT CODPROV,NOMBRE_PROV,NOMBRE_DEP FROM PROVINCIA P INNER JOIN

```

```

DEPARTAMENTO D

```

```

ON (P.CODDEP = D.CODDEP)

```

```

WHERE CODPROV LIKE CONCAT('%',BUS,'%') OR NOMBRE_PROV LIKE
CONCAT('%',BUS,'%');
ELSEIF TIPO='D' THEN
DELETE FROM PROVINCIA WHERE
CODPROV=BUS;
ELSEIF TIPO='E' THEN
SELECT * FROM PROVINCIA WHERE
CODPROV=BUS;
ELSEIF TIPO='L' THEN
SELECT CODPROV,NOMBRE_PROV FROM PROVINCIA
WHERE CODPROV=BUS OR NOMBRE_PROV LIKE CONCAT('%',BUS,'%');
END IF;
END$$
DELIMITER $$

```

```
CALL SP_BUSDEL_PROVINCIA('1','B')
```

```
SELECT * FROM PROVINCIA
```

```

DROP PROCEDURE IF EXISTS SP_ADDUP_PROVINCIA;
DELIMITER $$
CREATE PROCEDURE SP_ADDUP_PROVINCIA(CODPV INT,NOMPV
VARCHAR(40),CODDP INT,TIPO CHAR(1))
BEGIN
IF TIPO='G' THEN
INSERT INTO PROVINCIA VALUES(NULL,NOMPV,CODDP);
ELSEIF TIPO='A' THEN
UPDATE PROVINCIA SET NOMBRE_PROV=NOMPV, CODDEP=CODDP WHERE
CODPROV=CODPV;
END IF;
END$$
DELIMITER $$

```

```
CALL SP_ADDUP_PROVINCIA(NULL,'Huaylas',3,'G');
```

```

DROP PROCEDURE IF EXISTS SP_BUSDEL_DISTRITO;
DELIMITER $$
CREATE PROCEDURE SP_BUSDEL_DISTRITO
(
BUS VARCHAR(80),
TIPO CHAR(1)
)
BEGIN
IF TIPO='B' THEN
SELECT CPOSTAL,NOMBRE_DIST,NOMBRE_PROV FROM DISTRITO D INNER JOIN
PROVINCIA P
ON (D.CODPROV = P.CODPROV)
WHERE CPOSTAL LIKE CONCAT('%',BUS,'%') OR NOMBRE_DIST LIKE
CONCAT('%',BUS,'%');
ELSEIF TIPO='D' THEN
DELETE FROM DISTRITO WHERE
CPOSTAL=BUS;
ELSEIF TIPO='E' THEN
SELECT * FROM DISTRITO WHERE
CPOSTAL=BUS;
ELSEIF TIPO='L' THEN
SELECT CPOSTAL,NOMBRE_DIST FROM DISTRITO
WHERE CPOSTAL=BUS OR NOMBRE_DIST LIKE CONCAT('%',BUS,'%');
END IF;

```

```

END$$
DELIMITER $$

CALL SP_BUSDEL_DISTRITO('0271','B')

DROP PROCEDURE IF EXISTS SP_ADDUP_DISTRITO;
DELIMITER $$
CREATE PROCEDURE SP_ADDUP_DISTRITO(CPOS CHAR(5),NOMD VARCHAR(30),CPVC
INT,TIPO CHAR(1))
BEGIN
IF TIPO='G' THEN
INSERT INTO DISTRITO VALUES(CPOS,NOMD,CPVC);
ELSEIF TIPO='A' THEN
UPDATE DISTRITO SET NOMBRE_DIST=NOMD,CODPROV=CPVC WHERE
CPOSTAL=CPOS;
END IF;
END$$
DELIMITER $$

CALL SP_ADDUP_DISTRITO('09876','San Martin de Porres',1,'G');

SELECT * FROM DISTRITO;

DROP PROCEDURE IF EXISTS SP_BUSDEL_PERSONAL_ADMINISTRATIVO;
DELIMITER $$
CREATE PROCEDURE SP_BUSDEL_PERSONAL_ADMINISTRATIVO
(
BUS VARCHAR(80),
TIPO CHAR(1)
)
BEGIN
IF TIPO='B' THEN
SELECT CODPERS_ADMI,APELLIDOS,NOMBRES,DATE_FORMAT(FEC_NAC,'%d-%m-%Y')
AS FEC_NAC,DESCRIPCION_DOC_IDENT,NUM_DOC_IDENT,SEXO,
NOMBRE_DIST,DIRECCION,EMAIL_PERSONAL,EMAIL_INSTITUCIONAL,NOMBRE_AREA,
NOMBRE_CARGO,
ESTADO_PERSONAL,OBS_ESTADO_PERSONAL FROM PERSONAL_ADMINISTRATIVO PA
INNER JOIN TIPO_DOC_IDENTIDAD DI ON (PA.CODTIPO_DOC_IDENT =
DI.CODTIPO_DOC_IDENT)
INNER JOIN DISTRITO DS ON (PA.CPOSTAL = DS.CPOSTAL)
INNER JOIN AREA AR ON (PA.CODAREA = AR.CODAREA)
INNER JOIN CARGO CR ON (PA.CODCARGO = CR.CODCARGO)
WHERE ESTADO_PERSONAL = 'Activo' ORDER BY CODPERS_ADMI ASC;
ELSEIF TIPO='S' THEN
SELECT CODPERS_ADMI,APELLIDOS,NOMBRES,DATE_FORMAT(FEC_NAC,'%d-%m-%Y')
AS FEC_NAC,DESCRIPCION_DOC_IDENT,NUM_DOC_IDENT,SEXO,
NOMBRE_DIST,DIRECCION,EMAIL_PERSONAL,EMAIL_INSTITUCIONAL,NOMBRE_AREA,
NOMBRE_CARGO,
ESTADO_PERSONAL,OBS_ESTADO_PERSONAL FROM PERSONAL_ADMINISTRATIVO PA
INNER JOIN TIPO_DOC_IDENTIDAD DI ON (PA.CODTIPO_DOC_IDENT =
DI.CODTIPO_DOC_IDENT)
INNER JOIN DISTRITO DS ON (PA.CPOSTAL = DS.CPOSTAL)
INNER JOIN AREA AR ON (PA.CODAREA = AR.CODAREA)
INNER JOIN CARGO CR ON (PA.CODCARGO = CR.CODCARGO)
WHERE CODPERS_ADMI LIKE CONCAT('%',BUS,'%') OR APELLIDOS LIKE
CONCAT('%',BUS,'%')
OR NOMBRES LIKE CONCAT('%',BUS,'%') OR NUM_DOC_IDENT LIKE
CONCAT('%',BUS,'%')
OR ESTADO_PERSONAL LIKE CONCAT('%',BUS,'%')
ORDER BY CODPERS_ADMI ASC;

```



```

ELSEIF TIPO='D' THEN
DELETE FROM PERSONAL_ADMINISTRATIVO
WHERE CODPERS_ADMI=BUS;
ELSEIF TIPO='E' THEN
SELECT CODPERS_ADMI,APELLIDOS,NOMBRES,DATE_FORMAT(FEC_NAC,'%d-%m-%Y')
AS FEC_NAC,CODTIPO_DOC_IDENT,NUM_DOC_IDENT,
SEXO,CPOSTAL,DIRECCION,EMAIL_PERSONAL,EMAIL_INSTITUCIONAL,CODAREA,CODC
ARGO,ESTADO_PERSONAL,OBS_ESTADO_PERSONAL
FROM PERSONAL_ADMINISTRATIVO
WHERE CODPERS_ADMI=BUS;
ELSEIF TIPO='L' THEN
SELECT CODPERS_ADMI,CONCAT(APELLIDOS,' ',NOMBRES) FROM
PERSONAL_ADMINISTRATIVO
WHERE CODPERS_ADMI=BUS OR NOMBRES LIKE CONCAT('%',BUS,'%');
END IF;
END$$
DELIMITER $$

```

```
CALL SP_BUSDEL_PERSONAL_ADMINISTRATIVO('Inactivo','S');
```

```

DROP PROCEDURE IF EXISTS SP_ADDUP_PERSONAL_ADMINISTRATIVO;
DELIMITER $$
CREATE PROCEDURE SP_ADDUP_PERSONAL_ADMINISTRATIVO(CPA INT,APEPA
VARCHAR(40),NOMPA VARCHAR(40),FECNACPA DATE,
CTDI CHAR(5),NDIPA VARCHAR(16),SEXOPA VARCHAR(1),CPST CHAR(5),DIRPA
VARCHAR(60),EMAILPA VARCHAR(50),
EMAILINT VARCHAR(60),CODAR CHAR(4),CODCAR CHAR(5),ESTPA
VARCHAR(20),OBSEPA VARCHAR(200), TIPO CHAR(1))
BEGIN
IF TIPO='G' THEN
INSERT INTO PERSONAL_ADMINISTRATIVO
VALUES(NULL,APEPA,NOMPA,FECNACPA,CTDI,NDIPA,SEXOPA,CPST,DIRPA,EMAILPA,E
MAILINT,CODAR,CODCAR,ESTPA,OBSEPA);
ELSEIF TIPO='A' THEN
UPDATE PERSONAL_ADMINISTRATIVO SET
APELLIDOS=APEPA,NOMBRES=NOMPA,FEC_NAC=FECNACPA,
CODTIPO_DOC_IDENT=CTDI,NUM_DOC_IDENT=NDIPA,SEXO=SEXOPA,CPOSTAL=CPST,D
IRECCION=DIRPA,
EMAIL_PERSONAL=EMAILPA,EMAIL_INSTITUCIONAL=EMAILINT,CODAREA=CODAR,CO
DCARGO=CODCAR,
ESTADO_PERSONAL=ESTPA,OBS_ESTADO_PERSONAL=OBSEPA WHERE
CODPERS_ADMI=CPA;
END IF;
END$$
DELIMITER $$

```

```
CALL SP_ADDUP_PERSONAL_ADMINISTRATIVO(2,'Marquez Pacheco','Karina Melisa','2001-
04-11','TDI01','09876543','F','09876','Av. Arequipa
24','kari451@gmail.com','varciab@gmail.com','A001','C002','Activo','','A');
```

```

SELECT * FROM PERSONAL_ADMINISTRATIVO;
SELECT * FROM DISTRITO;
SELECT * FROM TIPO_DOC_IDENTIDAD;
SELECT * FROM AREA;
SELECT * FROM CARGO;

```

```
DROP PROCEDURE IF EXISTS SP_BUSDEL_NIVEL_ESTUDIO;
```

```

DELIMITER $$
CREATE PROCEDURE SP_BUSDEL_NIVEL_ESTUDIO
(
BUS VARCHAR(80),
TIPO CHAR(1)
)
BEGIN
IF TIPO='B' THEN
SELECT * FROM NIVEL_ESTUDIO WHERE
CODNIVEL_EST LIKE CONCAT('%',BUS,'%') OR
DESCRIPCION_NIVEL LIKE CONCAT('%',BUS,'%');
ELSEIF TIPO='D' THEN
DELETE FROM NIVEL_ESTUDIO WHERE
CODNIVEL_EST=BUS;
ELSEIF TIPO='E' THEN
SELECT * FROM NIVEL_ESTUDIO WHERE
CODNIVEL_EST=BUS;
END IF;
END$$
DELIMITER $$

```

```
CALL SP_BUSDEL_NIVEL_ESTUDIO('NE1','E')
```

```

DROP PROCEDURE IF EXISTS SP_ADDUP_NIVEL_ESTUDIO;
DELIMITER $$
CREATE PROCEDURE SP_ADDUP_NIVEL_ESTUDIO(CNE CHAR(3),DES VARCHAR(30),
TIPO CHAR(1))
BEGIN
IF TIPO='G' THEN
INSERT INTO NIVEL_ESTUDIO VALUES(CNE,DES);
ELSEIF TIPO='A' THEN
UPDATE NIVEL_ESTUDIO SET DESCRIPCION_NIVEL=DES WHERE CODNIVEL_EST=CNE;
END IF;
END$$
DELIMITER $$

```

```
CALL SP_ADDUP_NIVEL_ESTUDIO('','DSCDC','G');
```

```

DROP TRIGGER IF EXISTS GENERAR_CODIGO_NIVEL_ESTUDIO;
DELIMITER $$
CREATE TRIGGER GENERAR_CODIGO_NIVEL_ESTUDIO BEFORE INSERT ON
NIVEL_ESTUDIO FOR EACH ROW
BEGIN
DECLARE NEXT_CODE INT;
SET NEXT_CODE=(SELECT
IFNULL(MAX(CONVERT(SUBSTRING(CODNIVEL_EST,3),SIGNED INTEGER)),0) FROM
NIVEL_ESTUDIO)+1;
SET NEW.CODNIVEL_EST=CONCAT('NE',NEXT_CODE);
END$$
DELIMITER $$

```

```
SELECT * FROM NIVEL_ESTUDIO
```

```

DROP PROCEDURE IF EXISTS SP_BUSDEL_GRADO;
DELIMITER $$
CREATE PROCEDURE SP_BUSDEL_GRADO
(
BUS VARCHAR(80),

```

```

TIPO CHAR(1)
)
BEGIN
IF TIPO='B' THEN
SELECT CODGRADO,NOMBRE_GRADO,DESCRIPCION_NIVEL FROM GRADO GR INNER
JOIN NIVEL_ESTUDIO NE
ON (GR.CODNIVEL_EST = NE.CODNIVEL_EST)
WHERE CODGRADO LIKE CONCAT('%',BUS,'%') OR NOMBRE_GRADO LIKE
CONCAT('%',BUS,'%');
ELSEIF TIPO='D' THEN
DELETE FROM GRADO WHERE
CODGRADO=BUS;
ELSEIF TIPO='E' THEN
SELECT * FROM GRADO WHERE
CODGRADO=BUS;
ELSEIF TIPO='L' THEN
SELECT CODGRADO,CONCAT(NOMBRE_GRADO,',',DESCRIPCION_NIVEL) AS
NOMBRE_GRADO FROM GRADO GR INNER JOIN NIVEL_ESTUDIO NE
ON (GR.CODNIVEL_EST = NE.CODNIVEL_EST)
WHERE CODGRADO=BUS OR NOMBRE_GRADO LIKE CONCAT('%',BUS,'%');
END IF;
END$$
DELIMITER $$

CALL SP_BUSDEL_GRADO(",L");

SELECT * FROM GRADO

DROP PROCEDURE IF EXISTS SP_ADDUP_GRADO;
DELIMITER $$
CREATE PROCEDURE SP_ADDUP_GRADO(CODG CHAR(3),NOMG VARCHAR(20),CODNE
CHAR(3),TIPO CHAR(1))
BEGIN
IF TIPO='G' THEN
INSERT INTO GRADO VALUES(CODG,NOMG,CODNE);
ELSEIF TIPO='A' THEN
UPDATE GRADO SET NOMBRE_GRADO=NOMG, CODNIVEL_EST=CODNE WHERE
CODGRADO=CODG;
END IF;
END$$
DELIMITER $$

CALL SP_ADDUP_GRADO('G03','3R0','NE1','G');

DROP TRIGGER IF EXISTS GENERAR_CODIGO_GRADO;
DELIMITER $$
CREATE TRIGGER GENERAR_CODIGO_GRADO BEFORE INSERT ON GRADO FOR
EACH ROW
BEGIN
DECLARE NEXT_CODE INT;
SET NEXT_CODE=(SELECT
IFNULL(MAX(CONVERT(SUBSTRING(CODGRADO,2),SIGNED INTEGER)),0) FROM
GRADO)+1;
SET NEW.CODGRADO=CONCAT('G',LPAD(NEXT_CODE,2,'0'));
END$$
DELIMITER $$

SELECT * FROM GRADO;

```

```

DROP PROCEDURE IF EXISTS SP_BUSDEL_CURSO;
DELIMITER $$
CREATE PROCEDURE SP_BUSDEL_CURSO
(
BUS VARCHAR(80),
TIPO CHAR(1)
)
BEGIN
IF TIPO='B' THEN
SELECT CODCURSO,NOMBRE_CURSO,DESCRIPCION_NIVEL FROM CURSO CR INNER
JOIN NIVEL_ESTUDIO NE
ON (CR.CODNIVEL_EST = NE.CODNIVEL_EST)
WHERE CODCURSO LIKE CONCAT('%',BUS,'%') OR NOMBRE_CURSO LIKE
CONCAT('%',BUS,'%');
ELSEIF TIPO='D' THEN
DELETE FROM CURSO WHERE
CODCURSO=BUS;
ELSEIF TIPO='E' THEN
SELECT * FROM CURSO WHERE
CODCURSO=BUS;
ELSEIF TIPO='L' THEN
SELECT CODCURSO,CONCAT(NOMBRE_CURSO,' - ',DESCRIPCION_NIVEL) CURSO FROM
CURSO CR INNER JOIN NIVEL_ESTUDIO NE
ON (CR.CODNIVEL_EST = NE.CODNIVEL_EST)
WHERE CODCURSO=BUS OR NOMBRE_CURSO LIKE CONCAT('%',BUS,'%');
END IF;
END$$
DELIMITER $$

```

```

CALL SP_BUSDEL_CURSO('','L');

```

```

SELECT * FROM CURSO

```

```

DROP PROCEDURE IF EXISTS SP_ADDUP_CURSO;
DELIMITER $$
CREATE PROCEDURE SP_ADDUP_CURSO(CODC CHAR(4),NOMC VARCHAR(40),CODNE
CHAR(3),TIPO CHAR(1))
BEGIN
IF TIPO='G' THEN
INSERT INTO CURSO VALUES(CODC,NOMC,CODNE);
ELSEIF TIPO='A' THEN
UPDATE CURSO SET NOMBRE_CURSO=NOMC, CODNIVEL_EST=CODNE WHERE
CODCURSO=CODC;
END IF;
END$$
DELIMITER $$

```

```

CALL SP_ADDUP_CURSO('CU04','HUHUH','NE1','A');

```

```

DROP TRIGGER IF EXISTS GENERAR_CODIGO_CURSO;
DELIMITER $$
CREATE TRIGGER GENERAR_CODIGO_CURSO BEFORE INSERT ON CURSO FOR EACH
ROW
BEGIN
DECLARE NEXT_CODE INT;
SET NEXT_CODE=(SELECT
IFNULL(MAX(CONVERT(SUBSTRING(CODCURSO,2),SIGNED INTEGER)),0) FROM
CURSO)+1;
SET NEW.CODCURSO=CONCAT('C',LPAD(NEXT_CODE,3,'0'));

```

END\$\$
DELIMITER \$\$

DROP PROCEDURE IF EXISTS SP_BUSDEL_ANNIO_ESCOLAR;
DELIMITER \$\$
CREATE PROCEDURE SP_BUSDEL_ANNIO_ESCOLAR
(
BUS VARCHAR(80),
TIPO CHAR(1)
)
BEGIN
IF TIPO='B' THEN
SELECT CODANNIO,ANNIO_ESCO,DATE_FORMAT(FEC_INICIO_CLASES,'%d-%m-%Y')AS
FEC_INICIO_CLASES, DATE_FORMAT(FEC_SALIDA_VACACIONAL,'%d-%m-%Y')AS
FEC_SALIDA_VACACIONAL,
DATE_FORMAT(FEC_INGRESO_VACACIONAL,'%d-%m-%Y') AS
FEC_INGRESO_VACACIONAL,DATE_FORMAT(FEC_TERMINO_CLASES,'%d-%m-%Y')AS
FEC_TERMINO_CLASES,CONCAT(APELLIDOS,',',NOMBRES) AS NOMBRES
FROM ANNIO_ESCOLAR AE INNER JOIN PERSONAL_ADMINISTRATIVO PA
ON (AE.CODPERS_ADMI = PA.CODPERS_ADMI)
WHERE CODANNIO=BUS OR ANNIO_ESCO LIKE CONCAT('%',BUS,'%');
ELSEIF TIPO='D' THEN
DELETE FROM ANNIO_ESCOLAR WHERE
CODANNIO=BUS;
ELSEIF TIPO='E' THEN
SELECT CODANNIO,ANNIO_ESCO,DATE_FORMAT(FEC_INICIO_CLASES,'%d-%m-%Y')AS
FEC_INICIO_CLASES, DATE_FORMAT(FEC_SALIDA_VACACIONAL,'%d-%m-%Y')AS
FEC_SALIDA_VACACIONAL,
DATE_FORMAT(FEC_INGRESO_VACACIONAL,'%d-%m-%Y')AS
FEC_INGRESO_VACACIONAL,DATE_FORMAT(FEC_TERMINO_CLASES,'%d-%m-%Y')AS
FEC_TERMINO_CLASES,CODPERS_ADMI
FROM ANNIO_ESCOLAR WHERE CODANNIO=BUS;
ELSEIF TIPO='L' THEN
SELECT CODANNIO,ANNIO_ESCO FROM ANNIO_ESCOLAR
WHERE CODANNIO=BUS OR ANNIO_ESCO LIKE CONCAT('%',BUS,'%');
END IF;
END\$\$
DELIMITER \$\$

CALL SP_BUSDEL_ANNIO_ESCOLAR(', 'B');

SELECT * FROM ANNIO_ESCOLAR

DROP PROCEDURE IF EXISTS SP_ADDUP_ANNIO_ESCOLAR;
DELIMITER \$\$
CREATE PROCEDURE SP_ADDUP_ANNIO_ESCOLAR(CODAE INT,NOMAE CHAR(4),FIC
DATE,FSV DATE,FIV DATE,FTC DATE,CODPA INT,TIPO CHAR(1))
BEGIN
IF TIPO='G' THEN
INSERT INTO ANNIO_ESCOLAR VALUES(NULL,NOMAE,FIC,FSV,FIV,FTC,CODPA);
ELSEIF TIPO='A' THEN
UPDATE ANNIO_ESCOLAR SET ANNIO_ESCO=NOMAE,
FEC_INICIO_CLASES=FIC,FEC_SALIDA_VACACIONAL=FSV,
FEC_INGRESO_VACACIONAL=FIV,FEC_TERMINO_CLASES=FTC,CODPERS_ADMI=CODPA
WHERE CODANNIO=CODAE;
END IF;
END\$\$
DELIMITER \$\$

```
CALL SP_ADDUP_ANNIO_ESCOLAR(NULL,'2019','2019-03-15','2019-06-15','2019-07-05','2019-12-02',1,'G');
```

```
SELECT * FROM ANNIO_ESCOLAR;
```

```
DROP PROCEDURE IF EXISTS SP_BUSDEL_SECCION;
DELIMITER $$
CREATE PROCEDURE SP_BUSDEL_SECCION
(
BUS VARCHAR(80),
TIPO CHAR(1)
)
BEGIN
IF TIPO='B' THEN
SELECT
CODSECCION,ANNIO_ESCO,DESCRIPCION_SECCION,CONCAT(NOMBRE_GRADO,'
',DESCRIPCION_NIVEL)AS NOMBRE_GRADO,NUM_MAX_ESTUDIANTES_SECCION
FROM SECCION SC INNER JOIN ANNIO_ESCOLAR AE ON (SC.CODANNIO =
AE.CODANNIO)
INNER JOIN GRADO GR ON (SC.CODGRADO = GR.CODGRADO)
INNER JOIN NIVEL_ESTUDIO NE ON(GR.CODNIVEL_EST=NE.CODNIVEL_EST)
WHERE CODSECCION LIKE CONCAT('%',BUS,'%') OR DESCRIPCION_SECCION LIKE
CONCAT('%',BUS,'%')
OR ANNIO_ESCO LIKE CONCAT('%',BUS,'%');
ELSEIF TIPO='D' THEN
DELETE FROM SECCION WHERE
CODSECCION=BUS;
ELSEIF TIPO='E' THEN
SELECT * FROM SECCION WHERE
CODSECCION=BUS;
ELSEIF TIPO='L' THEN
SELECT CODSECCION,CONCAT(NOMBRE_GRADO,' - ',DESCRIPCION_NIVEL,' -
',DESCRIPCION_SECCION)
FROM SECCION SC INNER JOIN GRADO GR ON ( SC.CODGRADO = GR.CODGRADO)
INNER JOIN NIVEL_ESTUDIO NE ON ( GR.CODNIVEL_EST = NE.CODNIVEL_EST)
WHERE CODSECCION LIKE CONCAT('%',BUS,'%') OR DESCRIPCION_SECCION LIKE
CONCAT('%',BUS,'%');
END IF;
END$$
DELIMITER $$
```

```
CALL SP_BUSDEL_SECCION('L');
```

```
SELECT * FROM SECCION;
SELECT * FROM NIVEL_ESTUDIO;
SELECT * FROM GRADO;
SELECT * FROM ANNIO_ESCOLAR;
```

```
DROP PROCEDURE IF EXISTS SP_ADDUP_SECCION;
DELIMITER $$
CREATE PROCEDURE SP_ADDUP_SECCION(IDSC INT,CODAE INT,DESCS
VARCHAR(20),CODGR CHAR(3),NUMXE INT,TIPO CHAR(1))
BEGIN
IF TIPO='G' THEN
INSERT INTO SECCION VALUES(NULL,CODAE,DESCS,CODGR,NUMXE);
```

```

ELSEIF TIPO='A' THEN
UPDATE SECCION SET CODANNIO=CODAE, DESCRIPCION_SECCION=DESCS,
CODGRADO=CODGR, NUM_MAX_ESTUDIANTES_SECCION=NUMXE
WHERE CODSECCION=IDSC;
END IF;
END$$
DELIMITER $$

CALL SP_ADDUP_SECCION(NULL,1,'C','G03',60,'G');

SELECT * FROM SECCION;
SELECT * FROM GRADO;
SELECT * FROM ANNIO_ESCOLAR;

DROP PROCEDURE IF EXISTS SP_BUSDEL_NACIONALIDAD ;
DELIMITER $$
CREATE PROCEDURE SP_BUSDEL_NACIONALIDAD
(
BUS VARCHAR(80),
TIPO CHAR(1)
)
BEGIN
IF TIPO='B' THEN
SELECT * FROM NACIONALIDAD WHERE
CODNAC LIKE CONCAT('%',BUS,'%') OR
DESCRIPCION_NAC LIKE CONCAT('%',BUS,'%');
ELSEIF TIPO='D' THEN
DELETE FROM NACIONALIDAD WHERE
CODNAC=BUS;
ELSEIF TIPO='E' THEN
SELECT * FROM NACIONALIDAD WHERE
CODNAC=BUS;
ELSEIF TIPO='L' THEN
SELECT * FROM NACIONALIDAD WHERE
CODNAC=BUS OR DESCRIPCION_NAC LIKE CONCAT('%',BUS,'%');
END IF;
END$$
DELIMITER $$

CALL SP_BUSDEL_NACIONALIDAD('',L')

DROP PROCEDURE IF EXISTS SP_ADDUP_NACIONALIDAD;
DELIMITER $$
CREATE PROCEDURE SP_ADDUP_NACIONALIDAD(CODN CHAR(5),DSCN
VARCHAR(40), TIPO CHAR(1))
BEGIN
IF TIPO='G' THEN
INSERT INTO NACIONALIDAD VALUES(CODN,DSCN);
ELSEIF TIPO='A' THEN
UPDATE NACIONALIDAD SET DESCRIPCION_NAC=DSCN WHERE CODNAC=CODN;
END IF;
END$$
DELIMITER $$

CALL SP_ADDUP_NACIONALIDAD('N0002','Argentino','G');

DROP TRIGGER IF EXISTS GENERAR_CODIGO_NACIONALIDAD;
DELIMITER $$

```

```

CREATE TRIGGER GENERAR_CODIGO_NACIONALIDAD BEFORE INSERT ON
NACIONALIDAD FOR EACH ROW
BEGIN
DECLARE NEXT_CODE INT;
SET NEXT_CODE=(SELECT IFNULL(MAX(CONVERT(SUBSTRING(CODNAC,4),SIGNED
INTEGER)),0) FROM NACIONALIDAD)+1;
SET NEW.CODNAC=CONCAT('N',LPAD(NEXT_CODE,4,'0'));
END$$
DELIMITER $$

```

```

SELECT * FROM NACIONALIDAD

```

```

DROP PROCEDURE IF EXISTS SP_BUSDEL_NIVEL_IDIOMA_EXTRANJERO ;
DELIMITER $$

```

```

CREATE PROCEDURE SP_BUSDEL_NIVEL_IDIOMA_EXTRANJERO
(
BUS VARCHAR(80),
TIPO CHAR(1)
)

```

```

BEGIN
IF TIPO='B' THEN
SELECT * FROM NIVEL_IDIOMA_EXTRANJERO WHERE
CODNIVEL = BUS OR
DESCRIPCION_NIVEL LIKE CONCAT('%',BUS,'%');
ELSEIF TIPO='D' THEN
DELETE FROM NIVEL_IDIOMA_EXTRANJERO WHERE
CODNIVEL=BUS;
ELSEIF TIPO='E' THEN
SELECT * FROM NIVEL_IDIOMA_EXTRANJERO WHERE
CODNIVEL=BUS;
END IF;
END$$
DELIMITER $$

```

```

CALL SP_BUSDEL_NIVEL_IDIOMA_EXTRANJERO('','B')

```

```

DROP PROCEDURE IF EXISTS SP_ADDUP_NIVEL_IDIOMA_EXTRANJERO;
DELIMITER $$
CREATE PROCEDURE SP_ADDUP_NIVEL_IDIOMA_EXTRANJERO(CODDN INT,DN
VARCHAR(20), TIPO CHAR(1))

```

```

BEGIN
IF TIPO='G' THEN
INSERT INTO NIVEL_IDIOMA_EXTRANJERO VALUES(NULL,DN);
ELSEIF TIPO='A' THEN
UPDATE NIVEL_IDIOMA_EXTRANJERO SET DESCRIPCION_NIVEL=DN WHERE
CODNIVEL=CODDN;
END IF;
END$$
DELIMITER $$

```

```

CALL SP_ADDUP_NIVEL_IDIOMA_EXTRANJERO(2,'Intermed','A');

```

```

SELECT * FROM NIVEL_IDIOMA_EXTRANJERO;

```

```

DROP PROCEDURE IF EXISTS SP_BUSDEL_FRECUENCIA_DIA;
DELIMITER $$
CREATE PROCEDURE SP_BUSDEL_FRECUENCIA_DIA
(
BUS VARCHAR(80),

```



```

TIPO CHAR(1)
)
BEGIN
IF TIPO='B' THEN
SELECT * FROM FRECUENCIA_DIA WHERE
CODFREC = BUS OR
DESCRIPCION_DIA LIKE CONCAT('%',BUS,'%');
ELSEIF TIPO='D' THEN
DELETE FROM FRECUENCIA_DIA WHERE
CODFREC=BUS;
ELSEIF TIPO='E' THEN
SELECT * FROM FRECUENCIA_DIA WHERE
CODFREC=BUS;
END IF;
END$$
DELIMITER $$

CALL SP_BUSDEL_FRECUENCIA_DIA('1','D')

DROP PROCEDURE IF EXISTS SP_ADDUP_FRECUENCIA_DIA;
DELIMITER $$
CREATE PROCEDURE SP_ADDUP_FRECUENCIA_DIA(IDFD INT,DSCFD VARCHAR(60),
TIPO CHAR(1))
BEGIN
IF TIPO='G' THEN
INSERT INTO FRECUENCIA_DIA VALUES(NULL,DSCFD);
ELSEIF TIPO='A' THEN
UPDATE FRECUENCIA_DIA SET DESCRIPCION_DIA=DSCFD WHERE CODFREC=IDFD;
END IF;
END$$
DELIMITER $$

CALL SP_ADDUP_FRECUENCIA_DIA(2,'III','A');

DROP PROCEDURE IF EXISTS SP_BUSDEL_TIPO_AMBIENTE;
DELIMITER $$
CREATE PROCEDURE SP_BUSDEL_TIPO_AMBIENTE
(
BUS VARCHAR(80),
TIPO CHAR(1)
)
BEGIN
IF TIPO='B' THEN
SELECT * FROM TIPO_AMBIENTE WHERE
CODTIPO_AMB LIKE CONCAT('%',BUS,'%') OR
DESCRIPCION_AMBIENTE LIKE CONCAT('%',BUS,'%');
ELSEIF TIPO='D' THEN
DELETE FROM TIPO_AMBIENTE WHERE
CODTIPO_AMB=BUS;
ELSEIF TIPO='E' THEN
SELECT * FROM TIPO_AMBIENTE WHERE
CODTIPO_AMB=BUS;
ELSEIF TIPO='L' THEN
SELECT * FROM TIPO_AMBIENTE WHERE
CODTIPO_AMB=BUS OR DESCRIPCION_AMBIENTE LIKE CONCAT('%',BUS,'%');
END IF;
END$$
DELIMITER $$

CALL SP_BUSDEL_TIPO_AMBIENTE('','B')

```

```

DROP PROCEDURE IF EXISTS SP_ADDUP_TIPO_AMBIENTE;
DELIMITER $$
CREATE PROCEDURE SP_ADDUP_TIPO_AMBIENTE(CODTA CHAR(5),DSCTA
VARCHAR(40), TIPO CHAR(1))
BEGIN
IF TIPO='G' THEN
INSERT INTO TIPO_AMBIENTE VALUES(CODTA,DSCTA);
ELSEIF TIPO='A' THEN
UPDATE TIPO_AMBIENTE SET DESCRIPCION_AMBIENTE=DSCTA WHERE
CODTIPO_AMB=CODTA;
END IF;
END$$
DELIMITER $$

```

```

CALL SP_ADDUP_TIPO_AMBIENTE('T02','LaR','A');

```

```

DROP TRIGGER IF EXISTS GENERAR_CODIGO_TIPO_AMBIENTE;
DELIMITER $$
CREATE TRIGGER GENERAR_CODIGO_TIPO_AMBIENTE BEFORE INSERT ON
TIPO_AMBIENTE FOR EACH ROW
BEGIN
DECLARE NEXT_CODE INT;
SET NEXT_CODE=(SELECT
IFNULL(MAX(CONVERT(SUBSTRING(CODTIPO_AMB,2),SIGNED INTEGER)),0) FROM
TIPO_AMBIENTE)+1;
SET NEW.CODTIPO_AMB=CONCAT('T',LPAD(NEXT_CODE,2,'0'));
END$$
DELIMITER $$

```

```

DROP PROCEDURE IF EXISTS SP_BUSDEL_NIVEL_PISO;
DELIMITER $$
CREATE PROCEDURE SP_BUSDEL_NIVEL_PISO
(
BUS VARCHAR(80),
TIPO CHAR(1)
)
BEGIN
IF TIPO='B' THEN
SELECT * FROM NIVEL_PISO WHERE
CODNIVEL LIKE CONCAT('%',BUS,'%') OR
DESCRIPCION_NIVEL LIKE CONCAT('%',BUS,'%');
ELSEIF TIPO='D' THEN
DELETE FROM NIVEL_PISO WHERE
CODNIVEL=BUS;
ELSEIF TIPO='E' THEN
SELECT * FROM NIVEL_PISO WHERE
CODNIVEL=BUS;
ELSEIF TIPO='L' THEN
SELECT * FROM NIVEL_PISO WHERE
CODNIVEL=BUS OR DESCRIPCION_NIVEL LIKE CONCAT('%',BUS,'%');
END IF;
END$$
DELIMITER $$

```

```

CALL SP_BUSDEL_NIVEL_PISO('','B')

```

```

DROP PROCEDURE IF EXISTS SP_ADDUP_NIVEL_PISO;

```

```

DELIMITER $$
CREATE PROCEDURE SP_ADDUP_NIVEL_PISO(CODNP CHAR(3),DSCNP VARCHAR(40),
TIPO CHAR(1))
BEGIN
IF TIPO='G' THEN
INSERT INTO NIVEL_PISO VALUES(CODNP,DSCNP);
ELSEIF TIPO='A' THEN
UPDATE NIVEL_PISO SET DESCRIPCION_NIVEL=DSCNP WHERE CODNIVEL=CODNP;
END IF;
END$$
DELIMITER $$

```

```
CALL SP_ADDUP_NIVEL_PISO('N02','2DO Piso','G');
```

```

DROP TRIGGER IF EXISTS GENERAR_CODIGO_NIVEL_PISO;
DELIMITER $$
CREATE TRIGGER GENERAR_CODIGO_NIVEL_PISO BEFORE INSERT ON NIVEL_PISO
FOR EACH ROW
BEGIN
DECLARE NEXT_CODE INT;
SET NEXT_CODE=(SELECT IFNULL(MAX(CONVERT(SUBSTRING(CODNIVEL,2),SIGNED
INTEGER)),0) FROM NIVEL_PISO)+1;
SET NEW.CODNIVEL=CONCAT('N',LPAD(NEXT_CODE,2,'0'));
END$$
DELIMITER $$

```

```

DROP PROCEDURE IF EXISTS SP_BUSDEL_AMBIENTE;
DELIMITER $$
CREATE PROCEDURE SP_BUSDEL_AMBIENTE
(
BUS VARCHAR(80),
TIPO CHAR(1)
)
BEGIN
IF TIPO='B' THEN
SELECT CODAMBIENTE,
DESCRIPCION_AMBIENTE,DESCRIPCION_NIVEL,DESCRIPCION_AULA
FROM AMBIENTE AB
INNER JOIN TIPO_AMBIENTE TA ON (AB.CODTIPO_AMB = TA.CODTIPO_AMB)
INNER JOIN NIVEL_PISO NP ON (AB.CODNIVEL = NP.CODNIVEL)
WHERE CODAMBIENTE LIKE CONCAT('%',BUS,'%') OR DESCRIPCION_AULA LIKE
CONCAT('%',BUS,'%');
ELSEIF TIPO='D' THEN
DELETE FROM AMBIENTE WHERE
CODAMBIENTE=BUS;
ELSEIF TIPO='E' THEN
SELECT * FROM AMBIENTE WHERE
CODAMBIENTE=BUS;
ELSEIF TIPO='L' THEN
SELECT CODAMBIENTE,DESCRIPCION_AULA FROM AMBIENTE
WHERE CODAMBIENTE=BUS OR DESCRIPCION_AULA LIKE CONCAT('%',BUS,'%');
END IF;
END$$
DELIMITER $$

```

```
CALL SP_BUSDEL_AMBIENTE('','B')
```

```

DROP PROCEDURE IF EXISTS SP_ADDUP_AMBIENTE;
DELIMITER $$

```

```

CREATE PROCEDURE SP_ADDUP_AMBIENTE(CODAMB CHAR(3),CODTAMB
CHAR(3),CODNIVP CHAR(3),DSCA VARCHAR(40),TIPO CHAR(1))
BEGIN
IF TIPO='G' THEN
INSERT INTO AMBIENTE VALUES(CODAMB,CODTAMB,CODNIVP,DSCA);
ELSEIF TIPO='A'THEN
UPDATE AMBIENTE SET
CODTIPO_AMB=CODTAMB,CODNIVEL=CODNIVP,DESCRIPCION_AULA=DSCA WHERE
CODAMBIENTE=CODAMB;
END IF;
END$$
DELIMITER $$

```

```

CALL SP_ADDUP_AMBIENTE('A01','T01','N01','01','A');

```

```

DROP TRIGGER IF EXISTS GENERAR_CODIGO_AMBIENTE;
DELIMITER $$
CREATE TRIGGER GENERAR_CODIGO_AMBIENTE BEFORE INSERT ON AMBIENTE
FOR EACH ROW
BEGIN
DECLARE NEXT_CODE INT;
SET NEXT_CODE=(SELECT
IFNULL(MAX(CONVERT(SUBSTRING(CODAMBIENTE,2),SIGNED INTEGER)),0) FROM
AMBIENTE)+1;
SET NEW.CODAMBIENTE=CONCAT('A',LPAD(NEXT_CODE,2,'0'));
END$$
DELIMITER $$

```

```

DROP PROCEDURE IF EXISTS SP_BUSDEL_APODERADO;
DELIMITER $$
CREATE PROCEDURE SP_BUSDEL_APODERADO
(
BUS VARCHAR(80),
TIPO CHAR(1)
)
BEGIN
IF TIPO='B' THEN
SELECT COD_APODERADO,APELLIDOS,NOMBRES,DATE_FORMAT(FEC_NAC,'%d-%m-
%Y') AS FEC_NAC,SEXO,
DESCRIPCION_NAC,DESCRIPCION_DOC_IDENT,NUM_DOC,EST_CIVIL,NOMBRE_DIST,DIR
ECCION,ESTADO_APODERADO
FROM APODERADO AP
INNER JOIN NACIONALIDAD NC ON (AP.CODNAC = NC.CODNAC)
INNER JOIN TIPO_DOC_IDENTIDAD TD ON (AP.CODTIPO_DOC_IDENT =
TD.CODTIPO_DOC_IDENT)
INNER JOIN DISTRITO DT ON (AP.CPOSTAL = DT.CPOSTAL)
WHERE ESTADO_APODERADO = 'Activo';
ELSEIF TIPO='S' THEN
SELECT COD_APODERADO,APELLIDOS,NOMBRES,DATE_FORMAT(FEC_NAC,'%d-%m-
%Y') AS FEC_NAC,SEXO,
DESCRIPCION_NAC,DESCRIPCION_DOC_IDENT,NUM_DOC,EST_CIVIL,NOMBRE_DIST,DIR
ECCION,ESTADO_APODERADO
FROM APODERADO AP
INNER JOIN NACIONALIDAD NC ON (AP.CODNAC = NC.CODNAC)
INNER JOIN TIPO_DOC_IDENTIDAD TD ON (AP.CODTIPO_DOC_IDENT =
TD.CODTIPO_DOC_IDENT)
INNER JOIN DISTRITO DT ON (AP.CPOSTAL = DT.CPOSTAL)
HAVING COD_APODERADO LIKE CONCAT('%',BUS,'%') OR APELLIDOS LIKE
CONCAT('%',BUS,'%') OR NOMBRES LIKE CONCAT('%',BUS,'%')

```

```

OR NUM_DOC LIKE CONCAT('%',BUS,'%') OR ESTADO_APODERADO LIKE
CONCAT('%',BUS,'%');
ELSEIF TIPO='D' THEN
DELETE FROM APODERADO WHERE
COD_APODERADO=BUS;
ELSEIF TIPO='E' THEN
SELECT COD_APODERADO,APELLIDOS,NOMBRES,DATE_FORMAT(FEC_NAC,'%d-%m-
%Y') AS FEC_NAC,SEXO,
CODNAC,CODTIPO_DOC_IDENT,NUM_DOC,EST_CIVIL,CPOSTAL,DIRECCION,ESTADO_AP
ODERADO
FROM APODERADO WHERE COD_APODERADO=BUS;
ELSEIF TIPO='L' THEN
SELECT COD_APODERADO,CONCAT(APELLIDOS,' ',NOMBRES) AS NOMBRES FROM
APODERADO
WHERE COD_APODERADO=BUS OR NOMBRES LIKE CONCAT('%',BUS,'%');
END IF;
END$$
DELIMITER $$

```

```
CALL SP_BUSDEL_APODERADO('Espí','S')
```

```

DROP PROCEDURE IF EXISTS SP_ADDUP_APODERADO;
DELIMITER $$
CREATE PROCEDURE SP_ADDUP_APODERADO(IDAP INT,APEAP VARCHAR(40),NOMAP
VARCHAR(30),FNAC DATE,SEX CHAR(1),
CODN CHAR(5),CTDI CHAR(5),NUMDOC VARCHAR(16),ESTC CHAR(1),CPST CHAR(5),
DIR VARCHAR(60),ESTAP VARCHAR(25),TIPO CHAR(1))
BEGIN
IF TIPO='G' THEN
INSERT INTO APODERADO
VALUES(NULL,APEAP,NOMAP,FNAC,SEX,CODN,CTDI,NUMDOC,ESTC,CPST,DIR,ESTAP);
ELSEIF TIPO='A' THEN
UPDATE APODERADO SET
APELLIDOS=APEAP,NOMBRES=NOMAP,FEC_NAC=FNAC,SEXO=SEX,CODNAC=CODN,CO
DTIPO_DOC_IDENT=CTDI,
NUM_DOC=NUMDOC,EST_CIVIL=ESTC,CPOSTAL=CPST,DIRECCION=DIR,ESTADO_APODE
RADO=ESTAP
WHERE COD_APODERADO=IDAP;
END IF;
END$$
DELIMITER $$

```

```
CALL SP_ADDUP_APODERADO(NULL,'Lopez Mendoza','Diego Eduardo','1988-03-
11','M','N0001','TDI01','01769346','C','15331','Av. Peru 645',1,'G');
```

```

SELECT * FROM NACIONALIDAD;
SELECT * FROM TIPO_DOC_IDENTIDAD;
SELECT * FROM DISTRITO;

```

```
SELECT * FROM APODERADO
```

```
/*SP -> AGENDA_TELEF_APODERADO*/
```

```

DROP PROCEDURE IF EXISTS SP_BUSDEL_AGENDA_TELEF_APODERADO;
DELIMITER $$
CREATE PROCEDURE SP_BUSDEL_AGENDA_TELEF_APODERADO
(

```

```

BUS VARCHAR(80),
TIPO CHAR(1)
)
BEGIN
IF TIPO='B' THEN
SELECT IDAGENDA,APELLIDOS,NOMBRES,NUM_TELEF,DESCRIPCION_TELEF
FROM AGENDA_TELEF_APODERADO AG
INNER JOIN APODERADO AP ON (AG.COD_APODERADO = AP.COD_APODERADO)
WHERE IDAGENDA=BUS OR APELLIDOS LIKE CONCAT('%',BUS,'%') OR NOMBRES LIKE
CONCAT('%',BUS,'%');
ELSEIF TIPO='D' THEN
DELETE FROM AGENDA_TELEF_APODERADO WHERE
IDAGENDA=BUS;
ELSEIF TIPO='E' THEN
SELECT * FROM AGENDA_TELEF_APODERADO WHERE IDAGENDA=BUS;
END IF;
END$$
DELIMITER $$

```

```
CALL SP_BUSDEL_AGENDA_TELEF_APODERADO(",B')
```

```

DROP PROCEDURE IF EXISTS SP_ADDUP_AGENDA_TELEF_APODERADO;
DELIMITER $$
CREATE PROCEDURE SP_ADDUP_AGENDA_TELEF_APODERADO(IDAG INT, IDAP
INT, NUMT VARCHAR(12), DSCT VARCHAR(40), TIPO CHAR(1))
BEGIN
IF TIPO='G' THEN
INSERT INTO AGENDA_TELEF_APODERADO VALUES(NULL, IDAP, NUMT, DSCT);
ELSEIF TIPO='A' THEN
UPDATE AGENDA_TELEF_APODERADO SET
COD_APODERADO=IDAP, NUM_TELEF=NUMT, DESCRIPCION_TELEF=DSCT WHERE
IDAGENDA=IDAG;
END IF;
END$$
DELIMITER $$

```

```
CALL SP_ADDUP_AGENDA_TELEF_APODERADO(NULL,2,'987837364','Celular','G');
```

```
/*SP -> EMAIL_APODERADO*/
```

```

DROP PROCEDURE IF EXISTS SP_BUSDEL_EMAIL_APODERADO;
DELIMITER $$
CREATE PROCEDURE SP_BUSDEL_EMAIL_APODERADO
(
BUS VARCHAR(80),
TIPO CHAR(1)
)
BEGIN
IF TIPO='B' THEN
SELECT IDREG_EMAIL,APELLIDOS,NOMBRES,EMAIL
FROM EMAIL_APODERADO EP
INNER JOIN APODERADO AP ON ( EP.COD_APODERADO = AP.COD_APODERADO)
WHERE IDREG_EMAIL=BUS OR APELLIDOS LIKE CONCAT('%',BUS,'%') OR NOMBRES
LIKE CONCAT('%',BUS,'%');
ELSEIF TIPO='D' THEN
DELETE FROM EMAIL_APODERADO WHERE
IDREG_EMAIL=BUS;
ELSEIF TIPO='E' THEN
SELECT * FROM EMAIL_APODERADO WHERE

```

```
IDREG_EMAIL=BUS;
END IF;
END$$
DELIMITER $$
```

```
CALL SP_BUSDEL_EMAIL_APODERADO('B')
```

```
DROP PROCEDURE IF EXISTS SP_ADDUP_EMAIL_APODERADO;
DELIMITER $$
CREATE PROCEDURE SP_ADDUP_EMAIL_APODERADO(IDE INT, IDAP INT, EM
VARCHAR(40), TIPO CHAR(1))
BEGIN
IF TIPO='G' THEN
INSERT INTO EMAIL_APODERADO VALUES(NULL, IDAP, EM);
ELSEIF TIPO='A' THEN
UPDATE EMAIL_APODERADO SET COD_APODERADO=IDAP, EMAIL=EM WHERE
IDREG_EMAIL=IDE;
END IF;
END$$
DELIMITER $$
```

```
CALL SP_ADDUP_EMAIL_APODERADO(NULL, 2, 'hurtadoc@gmail.com', 'G');
```

```
/*SP -> DOCENTE*/
```

```
DROP PROCEDURE IF EXISTS SP_BUSDEL_DOCENTE;
DELIMITER $$
CREATE PROCEDURE SP_BUSDEL_DOCENTE
(
BUS VARCHAR(80),
TIPO CHAR(1)
)
BEGIN
IF TIPO='B' THEN
SELECT DNI_DOCENTE, APELLIDOS, NOMBRES, DATE_FORMAT(FEC_NAC, '%d-%m-%Y')
AS FEC_NAC, SEXO,
DESCRIPCION_NAC, LENGUA_MATERNA, DESCRIPCION_IDIOMA, DESCRIPCION_NIVEL, N
OMBRE_DIST,
DIRECCION, EST_CIVIL, EMAIL_PERSONAL, EMAIL_INSTITUCIONAL, ESTADO_DOC
FROM DOCENTE DC
INNER JOIN NACIONALIDAD NC ON (DC.CODNAC = NC.CODNAC)
INNER JOIN IDIOMA_EXTRANJERO IE ON (DC.CODIDIOMA = IE.CODIDIOMA)
INNER JOIN NIVEL_IDIOMA_EXTRANJERO NI ON (DC.CODNIVEL = NI.CODNIVEL)
INNER JOIN DISTRITO DT ON (DC.CPOSTAL = DT.CPOSTAL)
WHERE ESTADO_DOC = 'Activo';
ELSEIF TIPO='S' THEN
SELECT DNI_DOCENTE, APELLIDOS, NOMBRES, DATE_FORMAT(FEC_NAC, '%d-%m-%Y')
AS FEC_NAC, SEXO,
DESCRIPCION_NAC, LENGUA_MATERNA, DESCRIPCION_IDIOMA, DESCRIPCION_NIVEL, N
OMBRE_DIST,
DIRECCION, EST_CIVIL, EMAIL_PERSONAL, EMAIL_INSTITUCIONAL, ESTADO_DOC
FROM DOCENTE DC
INNER JOIN NACIONALIDAD NC ON (DC.CODNAC = NC.CODNAC)
INNER JOIN IDIOMA_EXTRANJERO IE ON (DC.CODIDIOMA = IE.CODIDIOMA)
INNER JOIN NIVEL_IDIOMA_EXTRANJERO NI ON (DC.CODNIVEL = NI.CODNIVEL)
INNER JOIN DISTRITO DT ON (DC.CPOSTAL = DT.CPOSTAL)
WHERE DNI_DOCENTE LIKE CONCAT('%', BUS, '%') OR APELLIDOS LIKE
CONCAT('%', BUS, '%')
OR NOMBRES LIKE CONCAT('%', BUS, '%') OR ESTADO_DOC LIKE CONCAT('%', BUS, '%');
```

```

ELSEIF TIPO='D' THEN
DELETE FROM DOCENTE WHERE
DNI_DOCENTE=BUS;
ELSEIF TIPO='E' THEN
SELECT DNI_DOCENTE,APELLIDOS,NOMBRES,DATE_FORMAT(FEC_NAC,'%d-%m-%Y')
AS FEC_NAC,
SEXO,CODNAC,LENGUA_MATERNA,CODIDIOMA,CODNIVEL,CPOSTAL,DIRECCION,EST_
CIVIL,EMAIL_PERSONAL,
EMAIL_INSTITUCIONAL,ESTADO_DOC FROM DOCENTE WHERE
DNI_DOCENTE=BUS;
ELSEIF TIPO='L' THEN
SELECT DNI_DOCENTE,CONCAT(APELLIDOS,' ',NOMBRES) AS APELLIDOS FROM
DOCENTE
WHERE DNI_DOCENTE=BUS OR NOMBRES LIKE CONCAT('%',BUS,'%');
END IF;
END$$
DELIMITER $$

```

```
CALL SP_BUSDEL_DOCENTE('','S');
```

```

DROP PROCEDURE IF EXISTS SP_ADDUP_DOCENTE;
DELIMITER $$
CREATE PROCEDURE SP_ADDUP_DOCENTE( DNID CHAR(8),APED VARCHAR(40),NOMD
VARCHAR(40),FNAC DATE,SEX CHAR(1),CODN CHAR(5),
LMD VARCHAR(20),CIED CHAR(6),CND INTEGER,CPST CHAR(5),DIR VARCHAR(60),ECD
CHAR(1),
EMP VARCHAR(50),EMI VARCHAR(40),ESTDC VARCHAR(30),TIPO CHAR(1))
BEGIN
IF TIPO='G' THEN
INSERT INTO DOCENTE
VALUES(DNID,APED,NOMD,FNAC,SEX,CODN,LMD,CIED,CND,CPST,DIR,ECD,EMP,EMI,ES
TDC);
ELSEIF TIPO='A' THEN
UPDATE DOCENTE SET
APELLIDOS=APED,NOMBRES=NOMD,FEC_NAC=FNAC,SEXO=SEX,CODNAC=CODN,LENG
UA_MATERNA=LMD,
CODIDIOMA=CIED,CODNIVEL=CND,CPOSTAL=CPST,DIRECCION=DIR,EST_CIVIL=ECD,E
MAIL_PERSONAL=EMP,
EMAIL_INSTITUCIONAL=EMI,ESTADO_DOC=ESTDC
WHERE DNI_DOCENTE=DNID;
END IF;
END$$
DELIMITER $$

```

```

CALL SP_ADDUP_DOCENTE('12354726','Mendoza Herrera','Marcos Jose','1988-03-
23','M','N0001','Castellano','IE0001',2,'11111','Av.
mmm','C','mar@gmail.com','victoriab@gmail.com','Activo','A');

```

```

SELECT * FROM NACIONALIDAD;
SELECT * FROM IDIOMA_EXTRANJERO;
SELECT * FROM NIVEL_IDIOMA_EXTRANJERO;
SELECT * FROM DISTRITO;

```

```
/*SP -> DISPONIBILIDAD_HORARIA*/
```

```

DROP PROCEDURE IF EXISTS SP_BUSDEL_DISPONIBILIDAD_HORARIA;
DELIMITER $$
CREATE PROCEDURE SP_BUSDEL_DISPONIBILIDAD_HORARIA

```



```
(
BUS VARCHAR(80),
TIPO CHAR(1)
)
BEGIN
IF TIPO='B' THEN
SELECT
CODDISPONIBILIDAD,APELLIDOS,NOMBRES,DESCRIPCION_DIA,HORA_INGRESO,HORA_
SALIDA
FROM DISPONIBILIDAD_HORARIA DH
INNER JOIN DOCENTE DC ON (DH.DNI_DOCENTE = DC.DNI_DOCENTE)
INNER JOIN FRECUENCIA_DIA FD ON (DH.CODFREC = FD.CODFREC)
WHERE CODDISPONIBILIDAD LIKE CONCAT('%',BUS,'%') OR APELLIDOS LIKE
CONCAT('%',BUS,'%') OR NOMBRES LIKE CONCAT('%',BUS,'%');
ELSEIF TIPO='D' THEN
DELETE FROM DISPONIBILIDAD_HORARIA WHERE
CODDISPONIBILIDAD=BUS;
ELSEIF TIPO='E' THEN
SELECT * FROM DISPONIBILIDAD_HORARIA WHERE CODDISPONIBILIDAD=BUS;
END IF;
END$$
DELIMITER $$

CALL SP_BUSDEL_DISPONIBILIDAD_HORARIA(",'B')
```

```
DROP PROCEDURE IF EXISTS SP_ADDUP_DISPONIBILIDAD_HORARIA;
DELIMITER $$
CREATE PROCEDURE SP_ADDUP_DISPONIBILIDAD_HORARIA(IDD INT,DNID
CHAR(8),IDF INT,HI TIME,HS TIME,TIPO CHAR(1))
BEGIN
IF TIPO='G' THEN
INSERT INTO DISPONIBILIDAD_HORARIA VALUES(NULL,DNID,IDF,HI,HS);
ELSEIF TIPO='A' THEN
UPDATE DISPONIBILIDAD_HORARIA SET
DNI_DOCENTE=DNID,CODFREC=IDF,HORA_INGRESO=HI,HORA_SALIDA=HS WHERE
CODDISPONIBILIDAD=IDD;
END IF;
END$$
DELIMITER $$
```

```
SELECT * FROM DISPONIBILIDAD_HORARIA;
SELECT * FROM FRECUENCIA_DIA;
SELECT * FROM DOCENTE;
```

```
CALL SP_ADDUP_DISPONIBILIDAD_HORARIA(NULL,'06254726',2,'9','13','G');
```

```
/*SP -> IDIOMA_EXTRANJERO*/
```

```
DROP PROCEDURE IF EXISTS SP_BUSDEL_IDIOMA_EXTRANJERO ;
DELIMITER $$
CREATE PROCEDURE SP_BUSDEL_IDIOMA_EXTRANJERO
(
BUS VARCHAR(80),
TIPO CHAR(1)
)
BEGIN
IF TIPO='B' THEN
SELECT * FROM IDIOMA_EXTRANJERO WHERE
CODIDIOMA LIKE CONCAT('%',BUS,'%') OR
```

```

DESCRIPCION_IDIOMA LIKE CONCAT('%',BUS,'%');
ELSEIF TIPO='D' THEN
DELETE FROM IDIOMA_EXTRANJERO WHERE
CODIDIOMA=BUS;
ELSEIF TIPO='E' THEN
SELECT * FROM IDIOMA_EXTRANJERO WHERE
CODIDIOMA=BUS;
ELSEIF TIPO='L' THEN
SELECT * FROM IDIOMA_EXTRANJERO WHERE
CODIDIOMA=BUS OR DESCRIPCION_IDIOMA LIKE CONCAT('%',BUS,'%');
END IF;
END$$
DELIMITER $$

CALL SP_BUSDEL_IDIOMA_EXTRANJERO(",B')

DROP PROCEDURE IF EXISTS SP_ADDUP_IDIOMA_EXTRANJERO;
DELIMITER $$
CREATE PROCEDURE SP_ADDUP_IDIOMA_EXTRANJERO(CODIE CHAR(6),DSCIE
VARCHAR(40), TIPO CHAR(1))
BEGIN
IF TIPO='G' THEN
INSERT INTO IDIOMA_EXTRANJERO VALUES(CODIE,DSCIE);
ELSEIF TIPO='A' THEN
UPDATE IDIOMA_EXTRANJERO SET DESCRIPCION_IDIOMA=DSCIE WHERE
CODIDIOMA=CODIE;
END IF;
END$$
DELIMITER $$

CALL SP_ADDUP_IDIOMA_EXTRANJERO('IE0002','Portugues','G');

/*TRIGGER -> IDIOMA_EXTRANJERO*/
DROP TRIGGER IF EXISTS GENERAR_CODIGO_IDIOMA_EXTRANJERO;
DELIMITER $$
CREATE TRIGGER GENERAR_CODIGO_IDIOMA_EXTRANJERO BEFORE INSERT ON
IDIOMA_EXTRANJERO FOR EACH ROW
BEGIN
DECLARE NEXT_CODE INT;
SET NEXT_CODE=(SELECT
IFNULL(MAX(CONVERT(SUBSTRING(CODIDIOMA,3),SIGNED INTEGER)),0) FROM
IDIOMA_EXTRANJERO)+1;
SET NEW.CODIDIOMA=CONCAT('IE',LPAD(NEXT_CODE,4,'0'));
END$$
DELIMITER $$

/*SP -> AGENDA_TELEF_DOCENTE*/

DROP PROCEDURE IF EXISTS SP_BUSDEL_AGENDA_TELEF_DOCENTE;
DELIMITER $$
CREATE PROCEDURE SP_BUSDEL_AGENDA_TELEF_DOCENTE
(
BUS VARCHAR(80),
TIPO CHAR(1)
)
BEGIN
IF TIPO='B' THEN
SELECT IDAGENDA,APELLIDOS,NOMBRES,NUM_TELEF,DESCRIPCION_TELEF
FROM AGENDA_TELEF_DOCENTE AG

```

```

INNER JOIN DOCENTE DC ON (AG.DNI_DOCENTE = DC.DNI_DOCENTE)
WHERE IDAGENDA=BUS OR APELLIDOS LIKE CONCAT('%',BUS,'%') OR NOMBRES LIKE
CONCAT('%',BUS,'%');
ELSEIF TIPO='D' THEN
DELETE FROM AGENDA_TELEF_DOCENTE WHERE
IDAGENDA=BUS;
ELSEIF TIPO='E' THEN
SELECT * FROM AGENDA_TELEF_DOCENTE WHERE IDAGENDA=BUS;
END IF;
END$$
DELIMITER $$

```

```

CALL SP_BUSDEL_AGENDA_TELEF_DOCENTE('','B')

```

```

DROP PROCEDURE IF EXISTS SP_ADDUP_AGENDA_TELEF_DOCENTE;
DELIMITER $$
CREATE PROCEDURE SP_ADDUP_AGENDA_TELEF_DOCENTE(IDAG INT,DNID
CHAR(8),NUMT VARCHAR(12),DSCT VARCHAR(20),TIPO CHAR(1))
BEGIN
IF TIPO='G' THEN
INSERT INTO AGENDA_TELEF_DOCENTE VALUES(NULL,DNID,NUMT,DSCT);
ELSEIF TIPO='A' THEN
UPDATE AGENDA_TELEF_DOCENTE SET
DNI_DOCENTE=DNID,NUM_TELEF=NUMT,DESCRIPCION_TELEF=DSCT WHERE
IDAGENDA=IDAG;
END IF;
END$$
DELIMITER $$

```

```

CALL SP_ADDUP_AGENDA_TELEF_DOCENTE(1,'12354726','987837464','ddd','A');

```

```

/*SP -> ESTUDIANTE*/

```

```

DROP PROCEDURE IF EXISTS SP_BUSDEL_ESTUDIANTE;
DELIMITER $$
CREATE PROCEDURE SP_BUSDEL_ESTUDIANTE
(
BUS VARCHAR(80),
TIPO CHAR(1)
)
BEGIN
IF TIPO='B' THEN
SELECT
CODESTUDIANTE,APE_PATerno,APE_MATERNO,NOMBRES,DATE_FORMAT(FEC_NAC,'
%d-%m-%Y') AS FEC_NAC,SEXO,
DESCRIPCION_NAC,DESCRIPCION_DOC_IDENT,NUM_DOC_IDENTIDAD,LENGUA_MATER
NA,DESCRIPCION_IDIOMA,
DESCRIPCION_NIVEL,NOMBRE_DIST,URBANIZACION,DIRECCION,EMAIL_INSTITUCIONA
L,TIPO_DISCAPACIDAD,
PADRE_VIVE,MADRE_VIVE,GRADO_INSTRUCCION_PADRE,GRADO_INSTRUCCION_MAD
RE,LABORA,NUM_HORAS_LABORA,
FOTO,ESTADO_ESTUDIANTE,OBS_EST_EST
FROM ESTUDIANTE ES
INNER JOIN NACIONALIDAD NC ON (ES.CODNAC = NC.CODNAC)
INNER JOIN TIPO_DOC_IDENTIDAD TD ON (ES.CODTIPO_DOC_IDENT =
TD.CODTIPO_DOC_IDENT)
INNER JOIN IDIOMA_EXTRANJERO IE ON (ES.CODIDIOMA = IE.CODIDIOMA)
INNER JOIN NIVEL_IDIOMA_EXTRANJERO NI ON (ES.CODNIVEL = NI.CODNIVEL)

```

```

INNER JOIN DISTRITO DT ON (ES.CPOSTAL = DT.CPOSTAL)
WHERE ESTADO_ESTUDIANTE = 'Activo';
ELSEIF TIPO='S' THEN
SELECT
CODESTUDIANTE,APE_PATERNO,APE_MATERNO,NOMBRES,DATE_FORMAT(FEC_NAC,'
%d-%m-%Y') AS FEC_NAC,SEXO,
DESCRIPCION_NAC,DESCRIPCION_DOC_IDENT,NUM_DOC_IDENTIDAD,LENGUA_MATER
NA,DESCRIPCION_IDIOMA,
DESCRIPCION_NIVEL,NOMBRE_DIST,URBANIZACION,DIRECCION,EMAIL_INSTITUCIONA
L,TIPO_DISCAPACIDAD,
PADRE_VIVE,MADRE_VIVE,GRADO_INSTRUCCION_PADRE,GRADO_INSTRUCCION_MAD
RE,LABORA,NUM_HORAS_LABORA,
FOTO,ESTADO_ESTUDIANTE,OBS_EST_EST
FROM ESTUDIANTE ES
INNER JOIN NACIONALIDAD NC ON (ES.CODNAC = NC.CODNAC)
INNER JOIN TIPO_DOC_IDENTIDAD TD ON (ES.CODTIPO_DOC_IDENT =
TD.CODTIPO_DOC_IDENT)
INNER JOIN IDIOMA_EXTRANJERO IE ON (ES.CODIDIOMA = IE.CODIDIOMA)
INNER JOIN NIVEL_IDIOMA_EXTRANJERO NI ON (ES.CODNIVEL = NI.CODNIVEL)
INNER JOIN DISTRITO DT ON (ES.CPOSTAL = DT.CPOSTAL)
WHERE CODESTUDIANTE LIKE CONCAT('%',BUS,'%') OR APE_PATERNO LIKE
CONCAT('%',BUS,'%') OR APE_MATERNO LIKE CONCAT('%',BUS,'%')
OR NOMBRES LIKE CONCAT('%',BUS,'%') OR ESTADO_ESTUDIANTE LIKE
CONCAT('%',BUS,'%');
ELSEIF TIPO='D' THEN
DELETE FROM ESTUDIANTE WHERE
CODESTUDIANTE=BUS;
ELSEIF TIPO='E' THEN
SELECT
CODESTUDIANTE,APE_PATERNO,APE_MATERNO,NOMBRES,DATE_FORMAT(FEC_NAC,'
%d-%m-%Y') AS FEC_NAC,SEXO,
CODNAC,CODTIPO_DOC_IDENT,NUM_DOC_IDENTIDAD,LENGUA_MATERNA,CODIDIOM
A,CODNIVEL,CPOSTAL,URBANIZACION,
DIRECCION,EMAIL_INSTITUCIONAL,TIPO_DISCAPACIDAD,PADRE_VIVE,MADRE_VIVE,G
RADO_INSTRUCCION_PADRE,
GRADO_INSTRUCCION_MADRE,LABORA,NUM_HORAS_LABORA,FOTO,ESTADO_ESTUDI
ANTE,OBS_EST_EST
FROM ESTUDIANTE WHERE CODESTUDIANTE=BUS;
ELSEIF TIPO='L' THEN
SELECT CODESTUDIANTE,CONCAT(APE_PATERNO,' ',APE_MATERNO,' ',NOMBRES) AS
ESTUDIANTE FROM ESTUDIANTE
WHERE CODESTUDIANTE=BUS OR NOMBRES LIKE CONCAT('%',BUS,'%');
END IF;
END$$
DELIMITER $$

```

```

CALL SP_BUSDEL_ESTUDIANTE(',B');

```

```

DROP PROCEDURE IF EXISTS SP_PAGINATION_ESTUDIANTE;
DELIMITER $$
CREATE PROCEDURE SP_PAGINATION_ESTUDIANTE
(
IN _START INTEGER,
IN _LIMIT INTEGER
)
BEGIN
PREPARE STMT FROM
" SELECT CODESTUDIANTE, APE_PATERNO, APE_MATERNO, NOMBRES,
DATE_FORMAT(FEC_NAC,'%d-%m-%Y') AS FEC_NAC, SEXO, DESCRIPCION_NAC,
DESCRIPCION_DOC_IDENT, NUM_DOC_IDENTIDAD, LENGUA_MATERNA,

```

```

DESCRIPCION_IDIOMA, DESCRIPCION_NIVEL, NOMBRE_DIST, URBANIZACION,
DIRECCION, EMAIL_INSTITUCIONAL, TIPO_DISCAPACIDAD, PADRE_VIVE, MADRE_VIVE,
GRADO_INSTRUCCION_PADRE, GRADO_INSTRUCCION_MADRE, LABORA,
NUM_HORAS_LABORA, FOTO, ESTADO_ESTUDIANTE, OBS_EST_EST FROM ESTUDIANTE
ES INNER JOIN NACIONALIDAD NC ON (ES.CODNAC = NC.CODNAC) INNER JOIN
TIPO_DOC_IDENTIDAD TD ON (ES.CODTIPO_DOC_IDENT = TD.CODTIPO_DOC_IDENT)
INNER JOIN IDIOMA_EXTRANJERO IE ON (ES.CODIDIOMA = IE.CODIDIOMA) INNER JOIN
NIVEL_IDIOMA_EXTRANJERO NI ON (ES.CODNIVEL = NI.CODNIVEL) INNER JOIN
DISTRITO DT ON (ES.CPOSTAL = DT.CPOSTAL) WHERE ESTADO_ESTUDIANTE = 'Activo'
LIMIT ?,? ";
SET @START = _START;
SET @LIMIT = _LIMIT;
EXECUTE STMT USING @START, @LIMIT;
DEALLOCATE PREPARE STMT;
END $$
DELIMITER $$

```

```
CALL SP_PAGINATION_ESTUDIANTE('0','4');
```

```

DROP PROCEDURE IF EXISTS SP_ADDUP_ESTUDIANTE;
DELIMITER $$
CREATE PROCEDURE SP_ADDUP_ESTUDIANTE(CODEST VARCHAR(11), APAT
VARCHAR(30), AMAT VARCHAR(30), NOM VARCHAR(30), FNAC DATE, SEX
CHAR(1), CODN CHAR(5),
CTDI CHAR(5), NDI VARCHAR(16), LMAT VARCHAR(30), CIDI CHAR(6), CNI
INTEGER, CPST CHAR(5), URB VARCHAR(40),
DIR VARCHAR(60), EMI VARCHAR(40), TDISC VARCHAR(40), PVIVE CHAR(2), MVIVE
CHAR(2), GIP VARCHAR(40), GIM VARCHAR(40),
LAB CHAR(2), NUMHL INTEGER, FOT VARCHAR(60), ESTE VARCHAR(20), OBSE
VARCHAR(50), TIPO CHAR(1))
BEGIN
IF TIPO='G' THEN
INSERT INTO ESTUDIANTE
VALUES(CODEST, APAT, AMAT, NOM, FNAC, SEX, CODN, CTDI, NDI, LMAT, CIDI, CNI, CPST, UR
B, DIR, EMI, TDISC, PVIVE, MVIVE, GIP, GIM, LAB, NUMHL, FOT, ESTE, OBSE);
ELSEIF TIPO='A' THEN
UPDATE ESTUDIANTE SET
APE_PATERNO=APAT, APE_MATERNO=AMAT, NOMBRES=NOM, FEC_NAC=FNAC, SEXO=SE
X, CODNAC=CODN, CODTIPO_DOC_IDENT=CTDI,
NUM_DOC_IDENTIDAD=NDI, LENGUA_MATERNA=LMAT, CODIDIOMA=CIDI, CODNIVEL=C
NI, CPOSTAL=CPST, URBANIZACION=URB,
DIRECCION=DIR, EMAIL_INSTITUCIONAL=EMI, TIPO_DISCAPACIDAD=TDISC, PADRE_VIV
E=PVIVE, MADRE_VIVE=MVIVE,
GRADO_INSTRUCCION_PADRE=GIP, GRADO_INSTRUCCION_MADRE=GIM, LABORA=LAB,
NUM_HORAS_LABORA=NUMHL, FOTO=FOT,
ESTADO_ESTUDIANTE=ESTE, OBS_EST_EST=OBSE
WHERE CODESTUDIANTE=CODEST;
END IF;
END $$
DELIMITER $$

```

```
SELECT * FROM ESTUDIANTE
```

```

CALL SP_ADDUP_ESTUDIANTE
('01-87654321', 'Montes', 'Garcia', 'Juan Jose', '2001-03-23', 'M', 'N0001', 'TDI01', '87654321', 'Castellano',
'IE0001', 1, '15331', 'nnn', 'nnn', 'victoriab@gmail.com', 'Ninguna', 'Si', 'Si', 'Segundaria', 'Segundaria', 'Si',
12, 'est01.png', 'Activo', 'G');

```

```

SELECT * FROM NACIONALIDAD;
SELECT * FROM TIPO_DOC_IDENTIDAD;
SELECT * FROM IDIOMA_EXTRANJERO;
SELECT * FROM NIVEL_IDIOMA_EXTRANJERO;
SELECT * FROM DISTRITO;

```

```

DROP PROCEDURE IF EXISTS SP_BUSDEL_USUARIO_PERSONAL_ADMINISTRATIVO;
DELIMITER $$

```

```

CREATE PROCEDURE SP_BUSDEL_USUARIO_PERSONAL_ADMINISTRATIVO
(
BUS VARCHAR(80),
TIPO CHAR(1)
)

```

```

BEGIN

```

```

IF TIPO='B' THEN

```

```

SELECT CODPERS_ADMI,CODUSUARIO,PASSWORD,ESTADO_USUARIO

```

```

FROM USUARIO_PERSONAL_ADMINISTRATIVO WHERE

```

```

CODPERS_ADMI LIKE CONCAT('%',BUS,'%') OR CODUSUARIO LIKE CONCAT('%',BUS,'%');

```

```

ELSEIF TIPO='D' THEN

```

```

DELETE FROM USUARIO_PERSONAL_ADMINISTRATIVO WHERE

```

```

CODPERS_ADMI=BUS;

```

```

ELSEIF TIPO='E' THEN

```

```

SELECT * FROM USUARIO_PERSONAL_ADMINISTRATIVO WHERE

```

```

CODPERS_ADMI=BUS;

```

```

END IF;

```

```

END$$

```

```

DELIMITER $$

```

```

CALL SP_BUSDEL_USUARIO_PERSONAL_ADMINISTRATIVO('B');

```

```

DROP PROCEDURE IF EXISTS SP_ADDUP_USUARIO_PERSONAL_ADMINISTRATIVO;
DELIMITER $$

```

```

CREATE PROCEDURE SP_ADDUP_USUARIO_PERSONAL_ADMINISTRATIVO(CODPA
INT,CODPAU VARCHAR(16),PWRD VARCHAR(100),ESTPA VARCHAR(20), TIPO CHAR(1))

```

```

BEGIN

```

```

IF TIPO='G' THEN

```

```

INSERT INTO USUARIO_PERSONAL_ADMINISTRATIVO

```

```

VALUES(CODPA,CODPAU,PWRD,ESTPA);

```

```

ELSEIF TIPO='A' THEN

```

```

UPDATE USUARIO_PERSONAL_ADMINISTRATIVO SET

```

```

CODUSUARIO=CODPAU,PASSWORD=PWRD,ESTADO_USUARIO=ESTPA WHERE

```

```

CODPERS_ADMI=CODPA;

```

```

END IF;

```

```

END$$

```

```

DELIMITER $$

```

```

CALL

```

```

SP_ADDUP_USUARIO_PERSONAL_ADMINISTRATIVO(1,'kfernandez','kfernandez','Inactivo','A');

```

```

DROP PROCEDURE IF EXISTS SP_VALIDAR_USUARIO_PERSONAL_ADMINISTRATIVO;
DELIMITER $$

```

```

CREATE PROCEDURE SP_VALIDAR_USUARIO_PERSONAL_ADMINISTRATIVO

```

```

(

```

```

USU VARCHAR(16)

```

```

)

```

```

BEGIN

```

```

DECLARE UE INT;

```

```

DECLARE ST INT;

```

```

SET UE:=(SELECT COUNT(*) FROM USUARIO_PERSONAL_ADMINISTRATIVO WHERE
CODUSUARIO=USU);
SET ST:=(SELECT COUNT(*) FROM USUARIO_PERSONAL_ADMINISTRATIVO WHERE
CODUSUARIO=USU AND ESTADO_USUARIO='Activo');
IF UE=0 THEN
SELECT 'UI';
ELSEIF ST=0 THEN
SELECT 'EI';
ELSEIF (UE=1 AND ST=1) THEN
SELECT PASSWORD FROM USUARIO_PERSONAL_ADMINISTRATIVO WHERE
CODUSUARIO=USU AND ESTADO_USUARIO='Activo';
END IF;
END$$
DELIMITER $$

CALL SP_VALIDAR_USUARIO_PERSONAL_ADMINISTRATIVO('00000001');

SELECT * FROM USUARIO_PERSONAL_ADMINISTRATIVO;

DROP PROCEDURE IF EXISTS SP_BUSDEL_MATRÍCULA;
DELIMITER $$
CREATE PROCEDURE SP_BUSDEL_MATRÍCULA
(
BUS VARCHAR(80),
TIPO CHAR(1)
)
BEGIN
IF TIPO='B' THEN
SELECT MT.NUM_MATRÍCULA,AE.ANNIO_ESCO,CONCAT(ST.APE_PATERNO,'
',ST.APE_MATERNO,' ',ST.NOMBRES)
AS ESTUDIANTE,CONCAT(PA.APELLIDOS,' ',PA.NOMBRES) AS
PERSONAL_RESPONSABLE_MATRI,
CONCAT(AP.APELLIDOS,' ',AP.NOMBRES) AS
APODERADO,NE.DESCRIPCION_NIVEL,GR.NOMBRE_GRADO,
SC.DESCRIPCION_SECCION,DATE_FORMAT(MT.FEC_MAT,'%d-%m-%Y') AS
FEC_MAT,MT.ESTADO_MAT
FROM MATRÍCULA MT INNER JOIN ANNIO_ESCOLAR AE ON ( MT.CODANNIO =
AE.CODANNIO)
INNER JOIN ESTUDIANTE ST ON ( MT.CODESTUDIANTE = ST.CODESTUDIANTE)
INNER JOIN PERSONAL_ADMINISTRATIVO PA ON ( MT.CODPERS_ADMI =
PA.CODPERS_ADMI)
INNER JOIN APODERADO AP ON ( MT.COD_APODERADO = AP.COD_APODERADO)
INNER JOIN SECCION SC ON ( MT.CODSECCION = SC.CODSECCION)
INNER JOIN GRADO GR ON ( SC.CODGRADO = GR.CODGRADO)
INNER JOIN NIVEL_ESTUDIO NE ON ( GR.CODNIVEL_EST = NE.CODNIVEL_EST)
WHERE NUM_MATRÍCULA LIKE CONCAT('%',BUS,'%') OR ST.APE_PATERNO LIKE
CONCAT('%',BUS,'%')
OR ST.APE_MATERNO LIKE CONCAT('%',BUS,'%')OR ST.NOMBRES LIKE
CONCAT('%',BUS,'%');
ELSEIF TIPO='D' THEN
DELETE FROM MATRÍCULA WHERE NUM_MATRÍCULA=BUS;
ELSEIF TIPO='E' THEN
SELECT
NUM_MATRÍCULA,CODANNIO,CODESTUDIANTE,CODPERS_ADMI,COD_APODERADO,CO
DSECCION,
DATE_FORMAT(FEC_MAT,'%d-%m-%Y') AS FEC_MAT,ESTADO_MAT
FROM MATRÍCULA WHERE NUM_MATRÍCULA=BUS;
END IF;
END$$
DELIMITER $$

```

```
CALL SP_BUSDEL_MATRÍCULA('B')
```

```
DROP PROCEDURE IF EXISTS SP_ADDUP_MATRÍCULA;
DELIMITER $$
CREATE PROCEDURE SP_ADDUP_MATRÍCULA(IDMT INT, IDAE INT, CODES
VARCHAR(11), IDPA INT, IDAP INT, IDSC INT, FECMT DATETIME, ESTMT
VARCHAR(20), TIPO CHAR(1))
BEGIN
IF TIPO='G' THEN
INSERT INTO MATRÍCULA VALUES(NULL, IDAE, CODES, IDPA, IDAP, IDSC, FECMT, ESTMT);
ELSEIF TIPO='A' THEN
UPDATE MATRÍCULA SET
CODANIO=IDAE, CODESTUDIANTE=CODES, CODPERS_ADMI=IDPA, COD_APODERADO=I
DAP,
CODSECCION=IDSC, FEC_MAT=FECMT, ESTADO_MAT=ESTMT
WHERE NUM_MATRÍCULA=IDMT;
END IF;
END$$
DELIMITER $$
```

```
CALL SP_ADDUP_MATRÍCULA(1,1,'01-01234567',2,1,1,'2019-08-01','Matriculado','A');
```

```
DROP PROCEDURE IF EXISTS SP_BUSDEL_PROGRAMACION_HORARIO_CLASES;
DELIMITER $$
CREATE PROCEDURE SP_BUSDEL_PROGRAMACION_HORARIO_CLASES
(
BUS VARCHAR(80),
TIPO CHAR(1)
)
BEGIN
IF TIPO='B' THEN
SELECT
IDPROG_HC, NOMBRE_GRADO, DESCRIPCION_NIVEL, DESCRIPCION_SECCION, CONCAT(A
PELLIDOS, ' ', NOMBRES) AS DOCENTE,
NOMBRE_CURSO, DESCRIPCION_AMBIENTE, DESCRIPCION_AULA, DESCRIPCION_DIA, HO
RA_INICIO_CLASES,
HORA_TERMINO_CLASES
FROM PROGRAMACION_HORARIO_CLASES PH
INNER JOIN SECCION SC ON ( PH.CODSECCION = SC.CODSECCION)
INNER JOIN GRADO GR ON ( SC.CODGRADO = GR.CODGRADO)
INNER JOIN NIVEL_ESTUDIO NE ON ( GR.CODNIVEL_EST = NE.CODNIVEL_EST)
INNER JOIN DOCENTE DC ON ( PH.DNI_DOCENTE = DC.DNI_DOCENTE)
INNER JOIN CURSO CR ON ( PH.CODCURSO = CR.CODCURSO)
INNER JOIN TIPO_AMBIENTE TA ON (PH.CODTIPO_AMB = TA.CODTIPO_AMB)
INNER JOIN AMBIENTE AB ON (PH.CODAMBIENTE = AB.CODAMBIENTE)
INNER JOIN FRECUENCIA_DIA FD ON ( PH.CODFREC = FD.CODFREC)
WHERE IDPROG_HC LIKE CONCAT('%', BUS, '%') OR NOMBRE_GRADO LIKE
CONCAT('%', BUS, '%') OR APELLIDOS LIKE CONCAT('%', BUS, '%') OR
NOMBRES LIKE CONCAT('%', BUS, '%') OR DESCRIPCION_DIA LIKE CONCAT('%', BUS, '%')
ORDER BY IDPROG_HC;
ELSEIF TIPO='D' THEN
DELETE FROM PROGRAMACION_HORARIO_CLASES WHERE IDPROG_HC=BUS;
ELSEIF TIPO='E' THEN
SELECT * FROM PROGRAMACION_HORARIO_CLASES
WHERE IDPROG_HC=BUS;
END IF;
END$$
DELIMITER $$
```



```
CALL SP_BUSDEL_PROGRAMACION_HORARIO_CLASES('B');
```

```
DROP PROCEDURE IF EXISTS SP_ADDUP_PROGRAMACION_HORARIO_CLASES;
DELIMITER $$
CREATE PROCEDURE SP_ADDUP_PROGRAMACION_HORARIO_CLASES(IDPHC
INT,IDSECC INT,DNIDC CHAR(8),CODCURS CHAR(4),CODTA CHAR(3),CODAMB
CHAR(3),CODFRD INT,HIC TIME, HTC TIME,TIPO CHAR(1))
BEGIN
IF TIPO='G' THEN
INSERT INTO PROGRAMACION_HORARIO_CLASES
VALUES(NULL,IDSECC,DNIDC,CODCURS,CODTA,CODAMB,CODFRD,HIC, HTC);
ELSEIF TIPO='A'THEN
UPDATE PROGRAMACION_HORARIO_CLASES SET
CODSECCION=IDSECC,DNI_DOCENTE=DNIDC,CODCURSO=CODCURS,
CODTIPO_AMB=CODTA,CODAMBIENTE=CODAMB,CODFREC=CODFRD,
HORA_INICIO_CLASES=HIC,HORA_TERMINO_CLASES=HTC
WHERE IDPROG_HC=IDPHC;
END IF;
END$$
DELIMITER $$
```

```
CALL
SP_ADDUP_PROGRAMACION_HORARIO_CLASES('1','36142344','C001','T01','A01',1,'08:00:00',
13:00:00,'G');
```

```
DROP TRIGGER IF EXISTS GENERAR_CODIGO_PROGRAMACION_HORARIO_CLASES;
DELIMITER $$
CREATE TRIGGER GENERAR_CODIGO_PROGRAMACION_HORARIO_CLASES BEFORE
INSERT ON PROGRAMACION_HORARIO_CLASES FOR EACH ROW
BEGIN
DECLARE NEXT_CODE INT;
SET NEXT_CODE=(SELECT MAX(IDPROG_HC)FROM
PROGRAMACION_HORARIO_CLASES)+1;
SET NEW.IDPROG_HC=NEXT_CODE;
END$$
DELIMITER $$
```

```
DROP PROCEDURE IF EXISTS SP_BUSDEL_ASISTENCIA;
DELIMITER $$
CREATE PROCEDURE SP_BUSDEL_ASISTENCIA
(
BUS VARCHAR(80),
TIPO CHAR(1)
)
BEGIN
IF TIPO='B' THEN
SELECT NUM_ASISTENCIA,CONCAT(GR.NOMBRE_GRADO,' ',NE.DESCRIPCION_NIVEL,'
',SC.DESCRIPCION_SECCION) AS GRADO,
CONCAT(DC.APELLIDOS,' ',DC.NOMBRES) AS
DOCENTE,CR.NOMBRE_CURSO,TA.DESCRIPCION_AMBIENTE,
AB.DESCRIPCION_AULA,DESCRIPCION_DIA,DATE_FORMAT(FEC_ASISTENCIA,'%d-%m-
%Y') AS FEC_ASISTENCIA
FROM ASISTENCIA AST
INNER JOIN SECCION SC ON ( AST.CODSECCION = SC.CODSECCION)
INNER JOIN GRADO GR ON ( SC.CODGRADO = GR.CODGRADO)
INNER JOIN NIVEL_ESTUDIO NE ON (GR.CODNIVEL_EST = NE.CODNIVEL_EST)
INNER JOIN DOCENTE DC ON ( AST.DNI_DOCENTE = DC.DNI_DOCENTE)
INNER JOIN CURSO CR ON ( AST.CODCURSO = CR.CODCURSO)
INNER JOIN TIPO_AMBIENTE TA ON (AST.CODTIPO_AMB = TA.CODTIPO_AMB)
```

```

INNER JOIN AMBIENTE AB ON (AST.CODAMBIENTE = AB.CODAMBIENTE)
INNER JOIN FRECUENCIA_DIA FD ON ( AST.CODFREC = FD.CODFREC)
WHERE NUM_ASISTENCIA LIKE CONCAT('%',BUS,'%') OR
DC.APELLIDOS LIKE CONCAT('%',BUS,'%') OR
DC.NOMBRES LIKE CONCAT('%',BUS,'%') OR
FEC_ASISTENCIA LIKE CONCAT('%',BUS,'%');
ELSEIF TIPO='D' THEN
DELETE FROM ASISTENCIA WHERE
NUM_ASISTENCIA=BUS;
ELSEIF TIPO='E' THEN
SELECT * FROM ASISTENCIA WHERE
NUM_ASISTENCIA=BUS;
END IF;
END$$
DELIMITER $$

```

```

CALL SP_BUSDEL_ASISTENCIA("B')

```

```

DROP PROCEDURE IF EXISTS SP_ADDUP_ASISTENCIA;
DELIMITER $$
CREATE PROCEDURE SP_ADDUP_ASISTENCIA(NASIS INT,IDSECC INT,DNIDC
CHAR(8),CODCURS CHAR(4),CODTA CHAR(3),CODAMB CHAR(3),CODFRD INT,FASIS
DATE,TIPO CHAR(1))
BEGIN
IF TIPO='G' THEN
INSERT INTO ASISTENCIA
VALUES(NULL,IDSECC,DNIDC,CODCURS,CODTA,CODAMB,CODFRD,FASIS);
ELSEIF TIPO='A'THEN
UPDATE ASISTENCIA SET
CODSECCION=IDSECC,DNI_DOCENTE=DNIDC,CODCURSO=CODCURS,
CODTIPO_AMB=CODTA,CODAMBIENTE=CODAMB,CODFREC=CODFRD,
FEC_ASISTENCIA=FASIS
WHERE NUM_ASISTENCIA=NASIS;
END IF;
END$$
DELIMITER $$

```

```

CALL SP_ADDUP_ASISTENCIA(NULL,2,'00000001','C001','T01','A01',1,'2019-05-08','G');

```

```

/*SP -> DETALLE_ASISTENCIA*/

```

```

DROP PROCEDURE IF EXISTS SP_BUSDEL_DETALLE_ASISTENCIA;
DELIMITER $$
CREATE PROCEDURE SP_BUSDEL_DETALLE_ASISTENCIA
(
BUS VARCHAR(80),
TIPO CHAR(1)
)
BEGIN
IF TIPO='B' THEN
SELECT NUM_ASISTENCIA,CONCAT(APE_PATERNO,',',APE_MATERNO,',',NOMBRES) AS
ESTUDIANTE,
CASE ASISTIO WHEN 1 THEN 'SI' ELSE 'NO' END AS ASISTIO,
CASE TARDANZA WHEN 1 THEN 'SI' ELSE 'NO' END AS TARDANZA,
CASE FALTO WHEN 1 THEN 'SI' ELSE 'NO' END AS FALTO,

```

```

CASE JUSTIFICADO WHEN 1 THEN 'SI' ELSE 'NO' END AS JUSTIFICADO
FROM DETALLE_ASISTENCIA DA
INNER JOIN ESTUDIANTE ST ON ( DA.CODESTUDIANTE = ST.CODESTUDIANTE)
WHERE NUM_ASISTENCIA LIKE CONCAT('%',BUS,'%') OR APE_PATERNO LIKE
CONCAT('%',BUS,'%');
ELSEIF TIPO='M' THEN

SELECT ST.CODESTUDIANTE, CONCAT(ST.APE_PATERNO,' ',ST.APE_MATERNO,'
',ST.NOMBRES) AS ESTUDIANTE
FROM MATRÍCULA MT
INNER JOIN ESTUDIANTE ST ON ( MT.CODESTUDIANTE = ST.CODESTUDIANTE)
INNER JOIN SECCION SC ON ( MT.CODSECCION = SC.CODSECCION)
INNER JOIN GRADO GR ON ( SC.CODGRADO = GR.CODGRADO)
INNER JOIN NIVEL_ESTUDIO NE ON ( GR.CODNIVEL_EST = NE.CODNIVEL_EST)
WHERE ESTADO_ESTUDIANTE = 'Activo' AND ESTADO_MAT= 'Matriculado' AND
MT.CODSECCION LIKE CONCAT('%',BUS,'%');
ELSEIF TIPO='D' THEN
DELETE FROM DETALLE_ASISTENCIA WHERE NUM_ASISTENCIA=BUS;
ELSEIF TIPO='E' THEN
SELECT * FROM DETALLE_ASISTENCIA WHERE NUM_ASISTENCIA=BUS;
END IF;
END$$
DELIMITER $$

CALL SP_BUSDEL_DETALLE_ASISTENCIA(",M')

DROP PROCEDURE IF EXISTS SP_SEL_DETALLE_ASISTENCIA;
DELIMITER $$
CREATE PROCEDURE SP_SEL_DETALLE_ASISTENCIA
(
BUS VARCHAR(80),
TIPO CHAR(1)
)
BEGIN
SELECT NUM_ASISTENCIA, CODESTUDIANTE, ASISTIO, TARDANZA, FALTO, JUSTIFICADO
FROM DETALLE_ASISTENCIA DA
INNER JOIN ASISTENCIA ST ON ( DA.CODESTUDIANTE = ST.CODESTUDIANTE)
INNER JOIN ESTUDIANTE ST ON ( DA.NUM_ASISTENCIA = ST.CODESTUDIANTE)
WHERE NUM_ASISTENCIA LIKE CONCAT('%',BUS,'%') OR APE_PATERNO LIKE
CONCAT('%',BUS,'%');
END$$
DELIMITER $$

CALL SP_SEL_DETALLE_ASISTENCIA(",B')
DROP PROCEDURE IF EXISTS SP_ADDUP_DETALLE_ASISTENCIA;
DELIMITER $$
CREATE PROCEDURE SP_ADDUP_DETALLE_ASISTENCIA(NASI INT, CODES
VARCHAR(11), ASI CHAR(2), TAR CHAR(2), FAL CHAR(2), JUS CHAR(2), TIPO CHAR(1))
BEGIN
IF TIPO='G' THEN
INSERT INTO DETALLE_ASISTENCIA VALUES(NASI, CODES, ASI, TAR, FAL, JUS);
ELSEIF TIPO='A' THEN
UPDATE DETALLE_ASISTENCIA SET
CODESTUDIANTE=CODES, ASISTIO=ASI, TARDANZA=TAR, FALTO=FAL, JUSTIFICADO=JUS
WHERE NUM_ASISTENCIA=NASI;
END IF;
END$$
DELIMITER $$

CALL SP_ADDUP_DETALLE_ASISTENCIA('A');

```

```

DROP PROCEDURE IF EXISTS SP_ID_ASISTENCIA;
DELIMITER $$
CREATE PROCEDURE SP_ID_ASISTENCIA()
BEGIN
SELECT MAX(NUM_ASISTENCIA) AS NUM_ASISTENCIA FROM ASISTENCIA;
END$$
DELIMITER $$

CALL SP_ID_ASISTENCIA();

DROP PROCEDURE IF EXISTS SP_BUSDEL_NOTAS;
DELIMITER $$
CREATE PROCEDURE SP_BUSDEL_NOTAS
(
BUS VARCHAR(80),
TIPO CHAR(1)
)
BEGIN
IF TIPO='B' THEN
SELECT IDREG_NOTAS,CONCAT(DC.APELLIDOS,' ',DC.NOMBRES) AS DOCENTE,
CONCAT(ST.APE_PATERNO,' ',ST.APE_MATERNO,' ',ST.NOMBRES) AS ESTUDIANTE,
CONCAT(GR.NOMBRE_GRADO,' ',NE.DESCRIPCION_NIVEL,' ',SC.DESCRIPCION_SECCION)
AS GRADO,
CR.NOMBRE_CURSO,EV1_BI,EV2_BI,EV3_BI,PP_BI,EF_BI,
EV1_BII,EV2_BII,EV3_BII,PP_BII,EF_BII,
EV1_BIII,EV2_BIII,EV3_BIII,PP_BIII,EF_BIII,
EV1_BIV,EV2_BIV,EV3_BIV,PP_BIV,EF_BIV,PF,ESTADO_CURSO
FROM NOTAS NTS
INNER JOIN DOCENTE DC ON ( NTS.DNI_DOCENTE = DC.DNI_DOCENTE)
INNER JOIN ESTUDIANTE ST ON ( NTS.CODESTUDIANTE = ST.CODESTUDIANTE)
INNER JOIN SECCION SC ON ( NTS.CODSECCION = SC.CODSECCION)
INNER JOIN GRADO GR ON ( SC.CODGRADO = GR.CODGRADO)
INNER JOIN NIVEL_ESTUDIO NE ON (GR.CODNIVEL_EST = NE.CODNIVEL_EST)
INNER JOIN CURSO CR ON ( NTS.CODCURSO = CR.CODCURSO)
WHERE IDREG_NOTAS LIKE CONCAT('%',BUS,'%');
ELSEIF TIPO='E' THEN
SELECT * FROM NOTAS WHERE IDREG_NOTAS=BUS;
END IF;
END$$
DELIMITER $$
CALL SP_BUSDEL_NOTAS('1','E');
DROP PROCEDURE IF EXISTS SP_SELECT_NOTAS;
DELIMITER $$
CREATE PROCEDURE SP_SELECT_NOTAS
(
BUS_1 VARCHAR(80),
BUS_2 VARCHAR(80)
)
BEGIN
SELECT IDREG_NOTAS,CONCAT(ST.APE_PATERNO,' ',ST.APE_MATERNO,'
 ',ST.NOMBRES) AS ESTUDIANTE,
GR.NOMBRE_GRADO,NE.DESCRIPCION_NIVEL,SC.DESCRIPCION_SECCION,CR.NOMBRE_
CURSO,
EF_BI,EF_BII,EF_BIII,EF_BIV,PF
FROM NOTAS NTS
INNER JOIN ESTUDIANTE ST ON ( NTS.CODESTUDIANTE = ST.CODESTUDIANTE)
INNER JOIN SECCION SC ON ( NTS.CODSECCION = SC.CODSECCION)
INNER JOIN GRADO GR ON ( SC.CODGRADO = GR.CODGRADO)
INNER JOIN NIVEL_ESTUDIO NE ON (GR.CODNIVEL_EST = NE.CODNIVEL_EST)

```

```

INNER JOIN CURSO CR ON ( NTS.CODCURSO = CR.CODCURSO)
WHERE CR.CODCURSO LIKE CONCAT('%',BUS_1,'%') AND SC.CODSECCION LIKE
CONCAT('%',BUS_2,'%');
END$$
DELIMITER $$

CALL SP_SELECT_NOTAS('C001','1');
DROP PROCEDURE IF EXISTS SP_ADDUP_NOTAS;
DELIMITER $$
CREATE PROCEDURE SP_ADDUP_NOTAS(IDREG_N INT(11),DNI_D CHAR(8),COD_ES
VARCHAR(11),COD_SEC INT(11),COD_CUR CHAR(4),EV1BI INT(11),
EV2BI INT(11),EV3BI INT(11),PPBI INT(11),EFBI INT(11),EV1BII INT(11),EV2BII
INT(11),EV3BII INT(11),
PPBII INT(11),EFBII INT(11),EV1BIII INT(11),EV2BIII INT(11),EV3BIII INT(11),PPBIII
INT(11),
EFBIII INT(11),EV1BIV INT(11),EV2BIV INT(11),EV3BIV INT(11),PPBIV INT(11),EFBIV
INT(11),PF INT(11),
ESTADO_CUR VARCHAR(20),TIPO CHAR(1))
BEGIN
IF TIPO='G' THEN
INSERT INTO NOTAS
VALUES(NULL,DNI_D,COD_ES,COD_SEC,COD_CUR,EV1BI,EV2BI,EV3BI,PPBI,EFBI,EV1BII,
EV2BII,EV3BII,
PPBII,EFBII,EV1BIII,EV2BIII,EV3BIII,PPBIII,EFBIII,EV1BIV,EV2BIV,EV3BIV,PPBIV,EFBIV,PF,
ESTADO_CUR);
ELSEIF TIPO='A'THEN
UPDATE NOTAS SET DNI_DOCENTE = DNI_D,CODESTUDIANTE = COD_ES,CODSECCION
= COD_SEC,
CODCURSO = COD_CUR,EV1_BI = EV1BI,EV2_BI = EV2BI,EV3_BI =EV3BI,PP_BI =
PPBI,EF_BI = EFBI,
EV1_BII = EV1BII,EV2_BII = EV2BII,EV3_BII = EV3BII,PP_BII = PPBII,EF_BII =
EFBII,EV1_BIII = EV1BIII,
EV2_BIII = EV2BIII,EV3_BIII = EV3BIII,PP_BIII = PPBIII,EF_BIII = EFBIII,EV1_BIV =
EV1BIV,EV2_BIV = EV2BIV,
EV3_BIV = EV3BIV,PP_BIV = PPBIV,EF_BIV =EFBIV,PF =PF,ESTADO_CURSO =
ESTADO_CUR
WHERE IDREG_NOTAS = IDREG_N;
END IF;
END$$
DELIMITER $$
CALL SP_ADDUP_NOTAS(NULL, '11111112', '01-
00000077', 1, 'C001', 14, 15, 15, 15, 17, 14, 16, 15, 17, 12, 20, 16, 12, 20, 12, 16, 16, 17, 14, 16, 12, 'Activo', 'G');

```

Codificacion del Sitio Web

Programacion de lado del Servidor

Se utilizo el lenguaje de programación PHP con el tipo de Programacion

Orientado a Objetos (POO) elaboración de este trabajo de aplicación web.

MODELO (php)

```
<?php
```

```

class Distrito {

    private $cpost;
    private $nom_dist;
    private $cod_prov;

    public function __Distrito() {
    }
    public function _Distrito($cpost, $nom_dist, $cod_prov) {
        $this->cpost = $cpost;
        $this->nom_dist = $nom_dist;
        $this->cod_prov = $cod_prov;
    }

    public function getCpost() {
        return $this->cpost;
    }

    public function getNom_dist() {
        return $this->nom_dist;
    }
    public function getCod_prov() {
        return $this->cod_prov;
    }

    public function setCpost($cpost) {
        $this->cpost = $cpost;
    }

    public function setNom_dist($nom_dist) {
        $this->nom_dist = $nom_dist;
    }
    public function setCod_prov($cod_prov) {
        $this->cod_prov = $cod_prov;
    }

}

```

Modelo Controlador (php)

```
<?php
```

```

class DDistrito {

    private $listadst;

    public function getArraydst() {
        return $this->listadst;
    }

    public function getSizedst() {

```

```

        return count($this->listadst);
    }

    public function getItemdst($f) {
        $dst = new Distrito();
        $dst->setCpost($this->listadst[$f]["cpost"]);
        $dst->setNom_dist($this->listadst[$f]["nom_dist"]);
        $dst->setCod_prov($this->listadst[$f]["cod_prov"]);
        return $dst;
    }

    public function getListdst($bus) {
        try {
            $tipo = 'B';
            $con = new Conexion();
            $query = $con->getcon()->prepare("CALL
SP_BUSDEL_DISTRITO(?,?)");
            $query->bindValue(1, $bus);
            $query->bindValue(2, $tipo);
            $query->execute();
            foreach ($query as $row) {
                $this->listadst[] = array(
                    'cpost' => $row[0],
                    'nom_dist' => $row[1],
                    'cod_prov' => $row[2],
                );
            }
        } catch (Exception $ex) {
            echo "Error Lista:" . $ex->getMessage();
        }
    }

    public function getDeletedst($cod) {
        try {
            $tipo = 'D';
            $con = new Conexion();
            $pre = $con->getcon()->
                prepare("CALL SP_BUSDEL_DISTRITO(?,?)");
            $pre->bindValue(1, $cod);
            $pre->bindValue(2, $tipo);
            if ($pre->execute() > 0)
                echo "DEE";
            else
                echo "No se Elimino";
        } catch (Exception $ex) {
            echo "Error Eliminar:" . $ex->getMessage();
        }
    }

    public function getSelectdst($id) {
        try {
            $tipo = 'E';
            $con = new Conexion();
            $query = $con->getcon()->prepare("CALL
SP_BUSDEL_DISTRITO(?,?)");
            $query->bindValue(1, $id);
            $query->bindValue(2, $tipo);
            $query->execute();
            foreach ($query as $row) {
                $this->listadst[] = array(
                    'cpost' => $row[0],

```

```

        'nom_dist' => $row[1],
        'cod_prov' => $row[2],
    );
}
} catch (Exception $ex) {
    echo "Error Select:" . $ex->getMessage();
}
}
}

public function getSelectCbxDist($idp) {
    try {
        $tipo = 'L';
        $con = new Conexion();
        $query = $con->getcon()->prepare("CALL
SP_BUSDEL_DISTRITO(?,?)");
        $query->bindValue(1, $idp);
        $query->bindValue(2, $tipo);
        $query->execute();
        foreach ($query as $row) {
            $this->listadst[] = array(
                'cpost' => $row[0],
                'nom_dist' => $row[1],
            );
        }
    } catch (Exception $ex) {
        echo "Error Select:" . $ex->getMessage();
    }
}

public function getAdddst(Distrito $obj) {
    try {
        $tipo = "G";
        $con = new Conexion();
        $pre = $con->getcon()->prepare("CALL
SP_ADDUP_DISTRITO(?,?,?,?)");
        $pre->bindValue(1, $obj->getCpost());
        $pre->bindValue(2, $obj->getNom_dist());
        $pre->bindValue(3, $obj->getCod_prov());
        $pre->bindValue(4, $tipo);
        if ($pre->execute() > 0)
            echo 'DAE';
        else
            echo 'No se Registro';
    } catch (Exception $exc) {
        echo "Error Add:" . $exc->getMessage();
    }
}

public function getUpdst(Distrito $obj) {
    try {
        $tipo = "A";
        $con = new Conexion();
        $pre = $con->getcon()->prepare("CALL
SP_ADDUP_DISTRITO(?,?,?,?)");
        $pre->bindValue(1, $obj->getCpost());
        $pre->bindValue(2, $obj->getNom_dist());
        $pre->bindValue(3, $obj->getCod_prov());
        $pre->bindValue(4, $tipo);
        if ($pre->execute() > 0)
            echo 'DUE';
        else
            echo 'No se Actualizado';
    }
}

```



```

    } catch (Exception $exc) {
        echo "Error Ap:" . $exc->getMessage();
    }
}
}

```

SERVICIOS (php)

```

<?php
require_once '../controller/BD/Conexion.php';
require_once '../controller/DAO/DDistrito.php';
require_once '../controller/DAO/DProvincia.php';
require_once '../model/Distrito.php';
require_once '../model/Provincia.php';

$dst = new DDistrito();
$pvc = new DProvincia();

$tipo = isset($_GET["tipo"]) != '' ? $_GET["tipo"] : '';

if ($tipo == 'eli') {
    $cod = $_POST['id'];
    echo $dst->getDeletedst($cod);
}

if ($tipo == 'lisdst') {
    $dst->getListdst("");
    echo json_encode($dst->getArraydst());
}

if ($tipo == 'lispvc') {
    $pvc->getSelectCBX("");
    echo json_encode($pvc->getArraypv());
}

if ($tipo == 'bus') {
    $dato = $_POST['search'];
    $dst->getListdst($dato);
    echo json_encode($dst->getArraydst());
}

if ($tipo == 'selectdst') {
    $sel = $_POST['id'];
    $dst->getSelectdst($sel);
    echo json_encode($dst->getArraydst());
}

if ($tipo == 'add') {
    $cpost = $_POST['cpstl'];
    $nom_dist = $_POST['nomdst'];
    $cod_prov = $_POST['idpvc'];
    $ds = new Distrito();
    $ds->_Distrito($cpost, $nom_dist, $cod_prov);
    $dst->getAdddst($ds);
}

if ($tipo == 'up') {

```

```

    $cpost = $_POST['cpt'];
    $nom_dist = $_POST['nomds'];
    $cod_prov = $_POST['idpvc'];
    $ds = new Distrito();
    $ds->_Distrito($cpost, $nom_dist, $cod_prov);
    $dst->getUpdst($ds);
}

```

Programacion Por lado del Cliente

Se dio interactividad a la aplicacion con JavaScript, JQuery y el diseño con el framework Bootstrap.

VISTA

HTML5

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <meta name="author" content="Dashboard">
    <meta name="keyword" content="Dashboard, Bootstrap, Admin,
Template, Theme, Responsive, Fluid, Retina">
    <title>VICTORIA BARCIA B.</title>
    <!--Link css-->
    <?php include_once './modulos/href_css.php'; ?>
    <?php include_once './modulos/href_css1.php'; ?>

  </head>
  <body>

    <section id="container">

      <?php include_once './modulos/headerStart.php'; ?>

      <?php include_once './modulos/sidebarStart.php'; ?>

      <div class="modal modal-success fade" id="modal-dst">
        <div class="modal-dialog">
          <div class="modal-content">
            <div class="modal-header">
              <button type="button" class="close" data-
dismiss="modal" aria-label="Close">
                <span aria-
hidden="true">&times;</span></button>
              <h4 class="modal-title">Registrar distrito</h4>
            </div>
            <form id="frm-dst" autocomplete="off">
              <div class="modal-body">

```

```

        <div class="box-body">
            <div class="row">
                <!--FIL 1-->
                <div class="col-md-12">
                    <div class="form-group col-lg-2
col-md-2">
                        <label>C-Postal</label>
                        <input type="text"
maxlength="5" class="form-control" id="cpst1" autofocus="">
                    </div>
                    <div class="form-group col-lg-5
col-md-5">
                        <label>Distrito</label>
                        <input type="text"
class="form-control" id="nom-dst">
                    </div>
                    <div class="form-group col-lg-5
col-md-5">
                        <label>Provincia</label>
                        <select class="form-
control" id="id-pvc">
                            </select>
                        </div>
                    </div>
                </div>
                <div class="modal-footer">
                    <button type="button" class="btn
btn-outline pull-left btn-danger" data-dismiss="modal">Cancelar</button>
                    <button type="submit" class="btn
btn-outline btn btn-success">Guardar</button>
                </div>
            </div>
        </form>
    </div>
</div>
</div>
</div>
<div class="row mt">
    <div class="col-lg-12">
        <h3><i class="fa fa-angle-right"></i>
        <button type="button" data-toggle="modal"
data-target="#modal-dst" class="btn btn-success ">Registrar
distrito</button>
        </h3>
        <div class="content-panel">
            <div class="row-fluid">
                <div class="span6">
                    <div id="hidden-table-info_length"
class="dataTables_length">
                        <h4>
                            <i class="fa fa-angle-
right"></i> Lista de distrito
                        </h4>
                    </div>
                </div>
            </div>
        </div>
    </div>
    <div class="span6">
        <form id="frm-search">

```

```

        <div class="dataTable_filter"
id="hidden-table-info_filter">
        <label>Buscar: <input
autocomplete="off" type="text" name="search" id="search" aria-
controls="hidden-table-info">
        </label>
        </div>
    </form>
</div>
</div>

<table class="table table-bordered table-
striped table-condensed cf">
    <thead class="cf">
        <tr>
            <th>Código Postal</th>
            <th>Distrito</th>
            <th>Provincia</th>
            <th style="width:
50px;">Update</th>
            <th style="width:
50px;">Delete</th>
        </tr>
    </thead>
    <tbody id="table-dst">
        </tbody>
</table>
<div class="row-fluid">
    <div class="span6">
        <div class="dataTable_info"
id="hidden-table-info_info">Showing 1 to 10 of 57 entries</div>
        </div>
        <div class="span6">
            <div class="dataTable_paginate
paginate_bootstrap pagination">
                <ul>
                    <li class="prev disabled">
                        <a href="#">←
                    </li>
                    <li class="active">
                        <a href="#">1</a>
                    </li>
                    <li>
                        <a href="#">2</a>
                    </li>
                    <li class="next">
                        <a href="#">Next → </a>
                    </li>
                </ul>
            </div>
        </div>
    </div>
</div>
</section>
</div>
</div>
</div>
</section>

```

```

</section>

<!--
<div class="modal modal-success fade" id="modal-dst-edi">
  <div class="modal-dialog">
    <div class="modal-content">
      <div class="modal-header">
        <button type="button" class="close" data-
dismiss="modal" aria-label="Close">
          <span aria-
hidden="true">&times;</span></button>
        <h4 class="modal-title">Editar distrito</h4>
      </div>
      <form id="frm-dst-up" autocomplete="off">
        <div class="modal-body">

          <div class="box-body">
            <div class="row">
              <div class="col-md-12">
                <div class="form-group col-lg-2
col-md-2">
                  <label>C-Postal</label>
                  <input type="text"
class="form-control" id="cpt" disabled="">
                </div>
                <div class="form-group col-lg-5
col-md-5">
                  <label>Distrito</label>
                  <input type="text"
class="form-control" id="nom-ds">
                </div>
                <div class="form-group col-lg-5
col-md-5">
                  <label>Provincia</label>
                  <select class="form-
control" id="idpvc">

                    </select>
                  </div>
                </div>
              </div>
            </div>
            <div class="modal-footer">
              <button type="button" class="btn btn-
outline pull-left btn-danger" data-dismiss="modal">Cancelar</button>
              <button type="submit" class="btn btn-
outline btn btn-success">Guardar</button>
            </div>
          </form>
        </div>
      </div>
    </div>
  </div>

  <?php include_once './modulos/footerStart.php'; ?>

</section>

  <?php include_once './modulos/src_js.php'; ?>
<?php include_once './modulos/src_jsl.php'; ?>

```

```

        <script src="ajax/ADistrito.js" type="text/javascript"></script>
    </body>
</html>

```

JQuery

```

$(document).ready(function () {

    ListDst();
    ListPvc();

    //BUSQUEDA
    $('#search').keyup(function () {
        if ($('#search').val()) {
            let search = ($('#search').val());
            const urlB = '../Service/SDistrito.php?tipo=bus'
            $.ajax({
                url: urlB,
                data: {search: search},
                type: 'POST',
                success: function (response) {
                    if (!response.error) {
                        const data = JSON.parse(response);
                        if (data != null) {
                            let template = '';
                            data.forEach(dato => {
                                template += `<tr slectId="${dato.cpost}">
                                    <td>${dato.cpost}</td>
                                    <td>${dato.nom_dist}</td>
                                    <td>${dato.cod_prov}</td>
                                    <td>
                                        <a href="#"><button type="submit"
class="update-dst btn btn-primary btn-xs"><i class="fa fa-
pencil"></i></button></a>
                                        </td>
                                    <td>
                                        <a href="#"><button type="submit"
class="delete-dst btn btn-danger btn-xs"><i class="fa fa-trash-o
"></i></button></a>
                                    </td>
                                </tr>
                            `;
                            });
                            $('#table-dst').html(template);
                        } else {
                            alertListNull();
                        }
                    }
                }
            });
        } else {
            ListDst();
        }
    });

    //LISTAR DISTRITO
    function ListDst() {
        const urlpvc = '../Service/SDistrito.php?tipo=lisdst'
        $.ajax({
            url: urlpvc,

```

```

type: 'GET',
success: function (response) {
    const data = JSON.parse(response);
    let template = '';
    data.forEach(dato => {
        template += `<tr slectId="${dato.cpost}">
            <td>${dato.cpost}</td>
            <td>${dato.nom_dist}</td>
            <td>${dato.cod_prov}</td>
            <td>
                <a href="#"><button type="submit" class="update-dst
btn btn-primary btn-xs"><i class="fa fa-pencil"></i></button></a>
            </td>
            <td>
                <a href="#"><button type="submit" class="delete-dst
btn btn-danger btn-xs"><i class="fa fa-trash-o "></i></button></a>
            </td>
        </tr>
    `;
    });
    $('#table-dst').html(template);
}
});
}

```

```

//INSERTAR DISTRITO
$('#frm-dst').submit(e => {
    e.preventDefault();
    const urlI = '../Service/SDistrito.php?tipo=add';
    var cpstl = $.trim($('#cpstl').val());
    var nomdst = $.trim($('#nom-dst').val());
    var idpvc = $.trim($('#id-pvc').val());

    if (cpstl.length == "") {
        swal(
            'Ingresar codigo postal',
            'Campo obligatorio',
            'warning'
        );
    }

    } else if (cpstl.length !== 5) {
        swal(
            'El código postal',
            'Debe ser igual a 5 dígitos',
            'warning'
        );
    }

    } else if (isNaN(cpstl)) {
        swal(
            'El código postal',
            'Debe ser numérico',
            'warning'
        );
    }

    } else if (nomdst.length == "") {
        swal(
            'Ingresar distrito',
            'Campo obligatorio',
            'warning'
        );
    }
}

```

```

} else {
    const postData = {
        cpstl: cpstl,
        nomdst: nomdst,
        idpvc: idpvc,
    };
    $.post(urlI, postData, (response) => {
        if (response == 'DAE') {
            swal(
                'Éxito ',
                'Distrito registrado correctamente ',
                'success'
            );
            $('#frm-dst').trigger('reset');
            $('#frm-search').trigger('reset');
            ListDst();
        };
        } else {
            swal(
                'Error al registrar',
                'Inténtalo de nuevo por favor... ',
                'warning'
            );
            $('#frm-search').trigger('reset');
        }
    });
}
});

//SELECT UPDATE DISTRITO
$(document).on('click', '.update-dst', (e) => {
    const element =
$(this)[0].activeElement.parentElement.parentElement.parentElement;
    const id = $(element).attr('slectId');
    const urlS = '../Service/SDistrito.php?tipo=selectdst'
    $.post(urlS, {id: id}, (response) => {
        const dato = JSON.parse(response);

        $('#modal-dst-edi').modal('show');
        $('#frm-search').trigger('reset')

        $('#cpt').val(dato[0].cpost);
        $('#nom-ds').val(dato[0].nom_dist);
        $('#idpvc').val(dato[0].cod_prov);

        $('#cpt').prop('disabled', true);
        $('#nom-ds').prop('disabled', false);
        $('#idpvc').prop('disabled', false);
        ;
    });
    e.preventDefault();
});

//UPDATE DISTRITO
$('#frm-dst-up').submit(e => {
    e.preventDefault();
    const urlU = '../Service/SDistrito.php?tipo=up'
    var cpt = $.trim($('#cpt').val());
    var nomds = $.trim($('#nom-ds').val());

```



```

var idpvc = $.trim($('#idpvc').val());

if (cpt.length == "" || nomds.length == "") {
    swal(
        'Advertencia',
        'En este formulario solo está permitido editar
registros',
        'warning'
    );
} else {
    const postData = {
        cpt: cpt,
        nomds: nomds,
        idpvc: idpvc,
    };

$.post(urlU, postData, (response) => {
    if (response == 'DUE') {
        swal(
            'Éxito ',
            'Distrito actualizado correctamente ',
            'success'
        );
        $('#frm-dst-up').trigger('reset');
        $('#frm-search').trigger('reset');
        ListDst();

        $('#cpt').prop('disabled', true);
        $('#nom-ds').prop('disabled', true);
        $('#idpvc').prop('disabled', true);

    } else {
        swal(
            'Error al actualizado',
            'Inténtalo de nuevo por favor... ',
            'warning'
        );
        $('#frm-search').trigger('reset');
    }
});
});

//DELETE DISTRITO
$(document).on('click', '.delete-dst', (e) => {
    if (confirm('¿Estás seguro(a) que desea eliminar el registro?')) {
        const element =
$(this)[0].activeElement.parentElement.parentElement;
        const id = $(element).attr('slectId');
        console.log(id);

        const urlD = '../Service/SDistrito.php?tipo=eli'
$.post(urlD, {id: id}, (response) => {
    if (response == 'DEE') {
        swal(
            'Éxito ',
            'Distrito eliminado correctamente',
            'success'
        );
        $('#frm-search').trigger('reset');
        ListDst();
    }
});
}
});

```

```

        } else {
            swal(
                'Error al eliminar',
                'Registro en uso',
                'warning'
            );
            $('#frm-search').trigger('reset');
        }
    });
}
});

//LISTA CBX DPT
function ListPvc() {
    const urlpvc = '../Service/SDistrito.php?tipo=lispvc'
    $.ajax({
        url: urlpvc,
        type: 'GET',
        success: function (response) {
            const datadpt = JSON.parse(response);
            let templat = '';
            datadpt.forEach(dato => {
                templat += `<option
value="${dato.cod_prov}">${dato.nom_prov}</option>`
            });
            //INSERT
            $('#id-pvc').html(templat);
            //UPDATE
            $('#idpvc').html(templat);
        }
    });
}

//ALERTA BUSQUEDA NULL
function alertListNull() {
    var unique_id = $.gritter.add({
        title: 'Lo sentimos',
        text: 'Registro no encontrado',
        image: 'plugin/img/portfolio/ico-dusqueda.ico',
        sticky: false,
        time: 2000,
        class_name: 'my-sticky-class'
    });

    return false;
}
});

```

Pruebas del sitio web

Manual de Usuario Aplicación web Gestión Académica SIGEAC

Para ingresar a la aplicación web a realizar cualquier actividad o consultas, el usuario deberá ingresar su “Usuario” y “Contraseña” , existen cuatro modos de acceso según sea el caso:

- a. Administrador
- b. Docente
- c. Apoderado
- d. Estudiante

Para cualquier de los cuatro casos; será de carácter obligatorio rellenar según corresponda los dos campos de Usuario y contraseña. Luego daremos clic en Iniciar Sesión y tendremos acceso a la aplicación web. (en caso de no completar ambos campos no tendrá acceso).

Perfil: Administrador/secretaria

Abrir el navegador y colocar la siguiente dirección URL:

http://localhost/SisVictoriaBarciaBoniffati/view/VLogin_usuario_personal_administrativo.php

A continuación, ingresar su usuario y contraseña, después clic en iniciar sesión.



Figura 25. Inicio de Sesión del Administrador

MATRÍCULA

EL personal encargado del área de secretaría académica se encargará de **registrar la información del nuevo estudiante** y apoderado; en caso de **que el estudiante es antiguo y pasa de grado académico** solamente se realizara la **Actualización** correspondiente.

A continuación, se muestra un ejemplo para ambos casos .

Nueva Matrícula

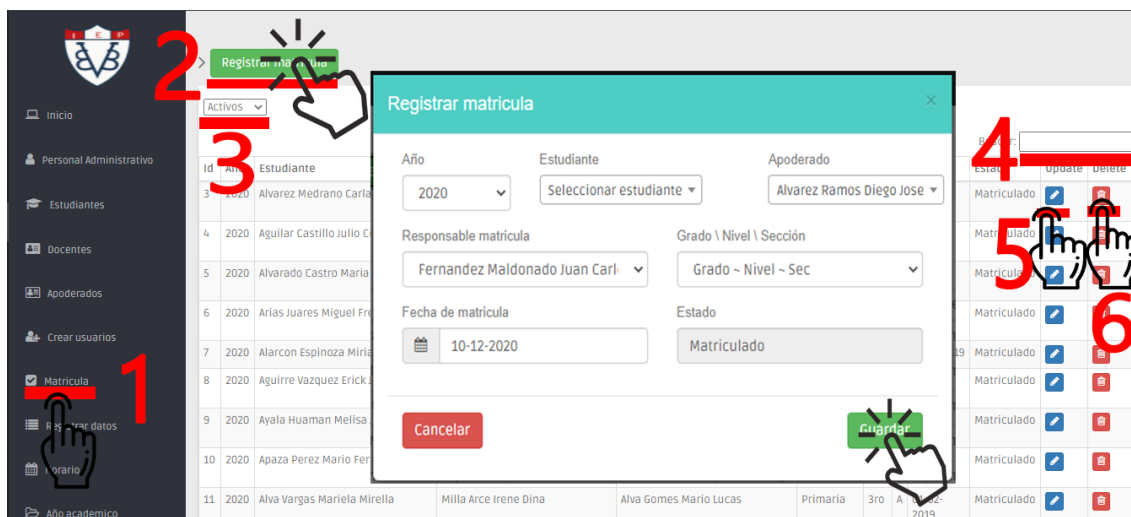


Figura 26. Inicio Para registro de Matricula

Realizaremos estos tres simples pasos para abrir la ventana de nueva matrícula:

1. Pulse la opción para acceder a la Matrícula.
2. Pulse la opción para acceder a la Asistencia
3. Pulse la opción para acceder a las Notas del estudiante.
4. Pulse la opción para asignar horario.
5. Pulse la opción para registrar los docentes.
6. Pulse la opción para registrar a los estudiantes.
7. Mensajes recientes.
8. Cerrar sesión.

The image shows a web form titled "Registrar matricula". It contains several dropdown menus and text input fields. The "Año" field is set to "2019". The "Estudiante" field is set to "Alvarez Medrano Carla Medrano". The "Responsable matricula" field is set to "Fernandez Maldonado Juan Carl". The "Apoderado" field is set to "Alvarez Ramos Diego Jose". The "Grado \ Nivel \ Sección" field is set to "Grado - Nivel - Sec". The "Fecha de matricula" field is empty. The "Estado" field is set to "Matriculado". At the bottom of the form, there are two buttons: "Cancelar" (red) and "Guardar" (green).

Figura 27. Registro de Matrícula

Todos los campos serán rellenados de manera obligatorio, una vez completados se le dará el estado de “Matriculado” y clic en Guardar. Automáticamente todos los datos ingresados serán almacenados en la base de datos de la Institución educativa. En caso de querer finalizar la inscripción por algún motivo o error solo daremos clic en cancelar.

Perfil: Docente

Abrir el navegador y colocar la siguiente dirección URL:

http://localhost/Sis-VictoriaBarciaBoniffati/view_doc/VLogin_usuario_docente.php

A continuación, ingresar su usuario y contraseña, después clic en iniciar sesión.

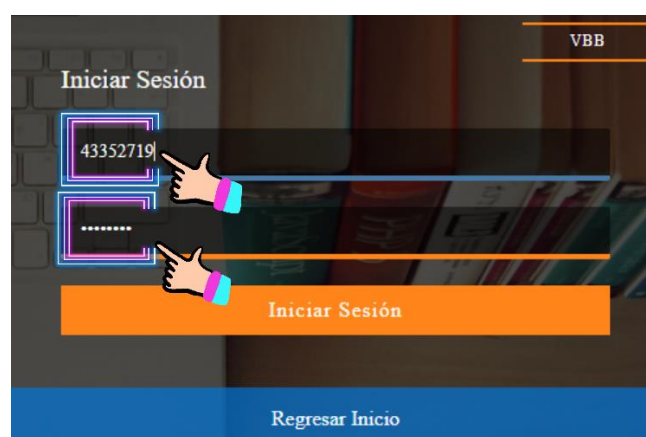


Figura 28. Inicio de Sesión del Docente

Funcionalidad General

La interfaz de la aplicación web se divide en dos zonas:

- **Menú Perfil Docente**
- **Menú Académico**

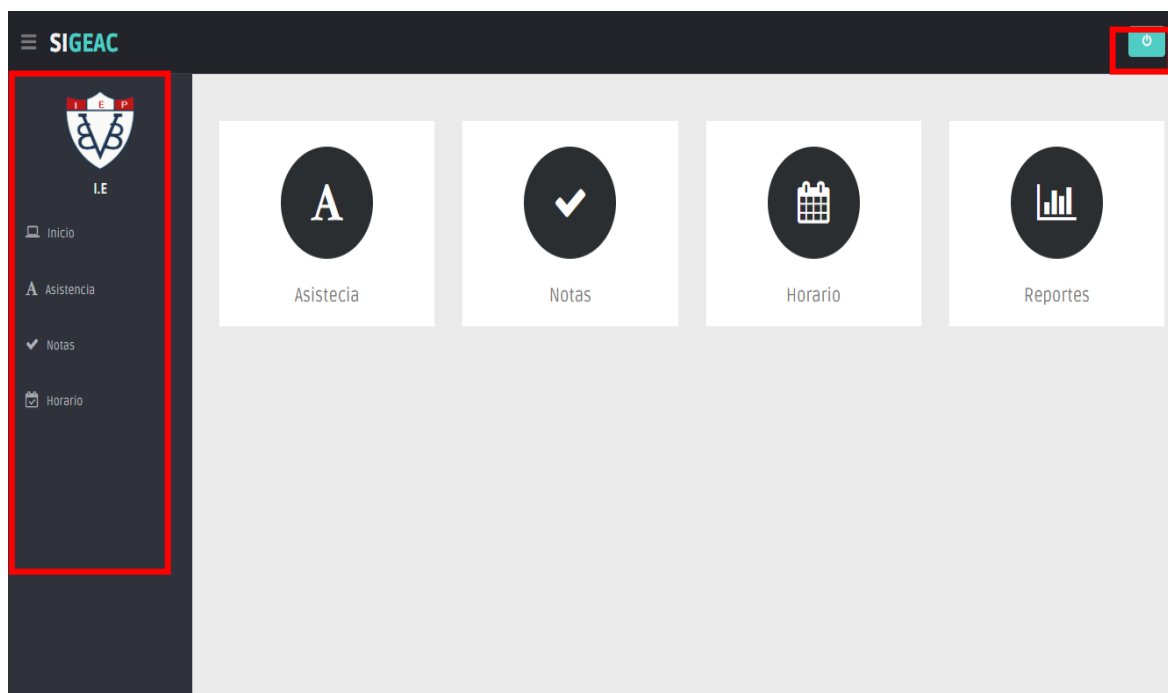


Figura 29. Menú Docente

Menú Perfil Docente

- **Cerrar Sesión**

Permite al Docente salir de la aplicación web de forma correcta y segura.

Para ello seleccionar con un clic izquierdo en el siguiente ícono:

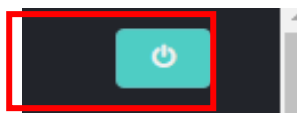


Figura 30. Botón de cerrar sesión

Muestra el siguiente anuncio y nuevamente dará un clic izquierdo en Cerrar para salir de manera segura de la aplicación web Institucional.

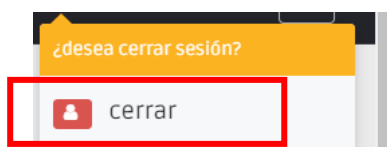


Figura 31. Cierre de sesión

▪ Menú Académico

▪ Inicio

Se mostrará en la parte derecha de la pantalla los siguientes botones de acceso:

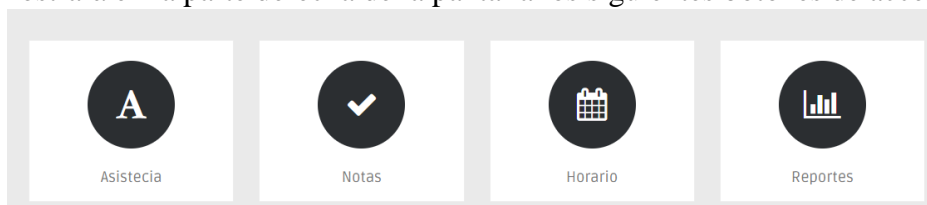


Figura 32. Botones de Acceso

▪ Asistencia

Muestra al docente la relación de estudiantes según el grado y el curso.

Grado	Nivel	Sección	Curso	Fecha	Estudiante	A	T	P	I	Edit.
1ro	Primaria	A	Matemática	11-12-2020	Alvarez Medrano Carla Melisa	si	no	no	no	[icon]
1ro	Primaria	A	Matemática	11-12-2020	Aguilar Castillo Julio Cesar	si	si	no	no	[icon]
1ro	Primaria	A	Matemática	11-12-2020	Alvarado Castro Maria Juana	si	no	no	no	[icon]
1ro	Primaria	A	Matemática	11-12-2020	Arias Juarez Miguel Freddy	no	no	si	si	[icon]

Figura 33. Relación de Estudiantes

Después de pasar lista a la hora indicada el docente dará clic en “Guardar”.

▪ Notas

En esta opción el docente podrá ingresar las notas obtenidas de los estudiantes según el grado y curso.

Figura 34. Tablero de Notas

Al finalizar el llenado de notas el docente deberá dar clic en “Guardar” para que todas las modificaciones queden grabadas correctamente. .

▪ Horario

Facilita al docente la programación de horarios de clases, según: día, hora, curso, tipo de ambiente.

Lunes	Viernes	Lunes	Jueves	Martes	Martes	Miércoles	Viernes	Miércoles	Jueves
08:00 10:30	08:00 10:30	11:00 13:00	11:00 13:00	08:00 10:30	11:30 13:00	08:00 10:30	11:00 13:00	11:00 13:00	08:00 10:30
Matemática 1ro Primaria A Aula 01	Matemática 1ro Primaria A Aula 01	Comunicación 1ro Primaria A Aula 01	Comunicación 1ro Primaria A Aula 01	Personal social 1ro Primaria A Aula 01	Arte y cultura 1ro Primaria A Aula 01	Ciencia y tecnología 1ro Primaria A Aula 01	Ciencia y tecnología 1ro Primaria A Aula 01	Educación física 1ro Primaria A Aula 01	Educación religiosa 1ro Primaria A Aula 01

Figura 35. Tablero de Horarios

Perfil Estudiante

Abrir su navegador de preferencia e ingrese la siguiente dirección URL:

http://localhost/Sis-VictoriaBarciaBoniffati/view_est/VLogin_usuario_estudiante.php

Después ingresar su Usuario y Contraseña ,

Es de carácter obligatorio el llenado de ambos campos de Usuario y contraseña. Luego daremos clic en Iniciar Sesión y tendremos acceso a la aplicación web. (en caso de no completar ambos campos no tendrá acceso).

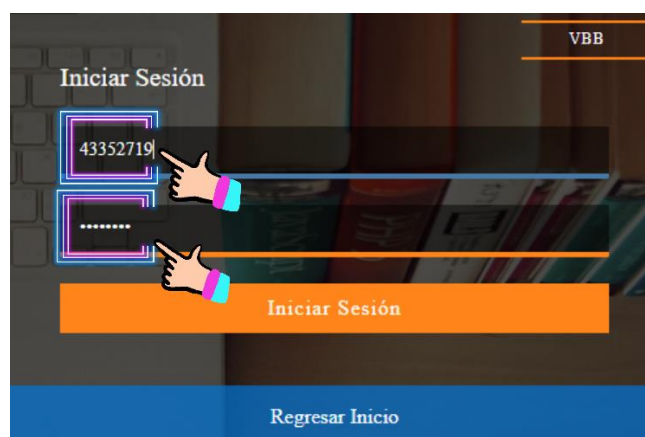


Figura 36. Inicio de Sesión Estudiante

Ejemplo:

Usuario: DNI del estudiante.

Contraseña: Ingresar la contraseña otorgada por el administrador de la I.E.

Iniciar Sesión: Proceso de ingreso al menú principal de la aplicación web modo estudiante.

Funcionalidad General

La interfaz de la aplicación web se divide en dos zonas:

- **Menú Perfil Estudiante**
- **Menú Académico**

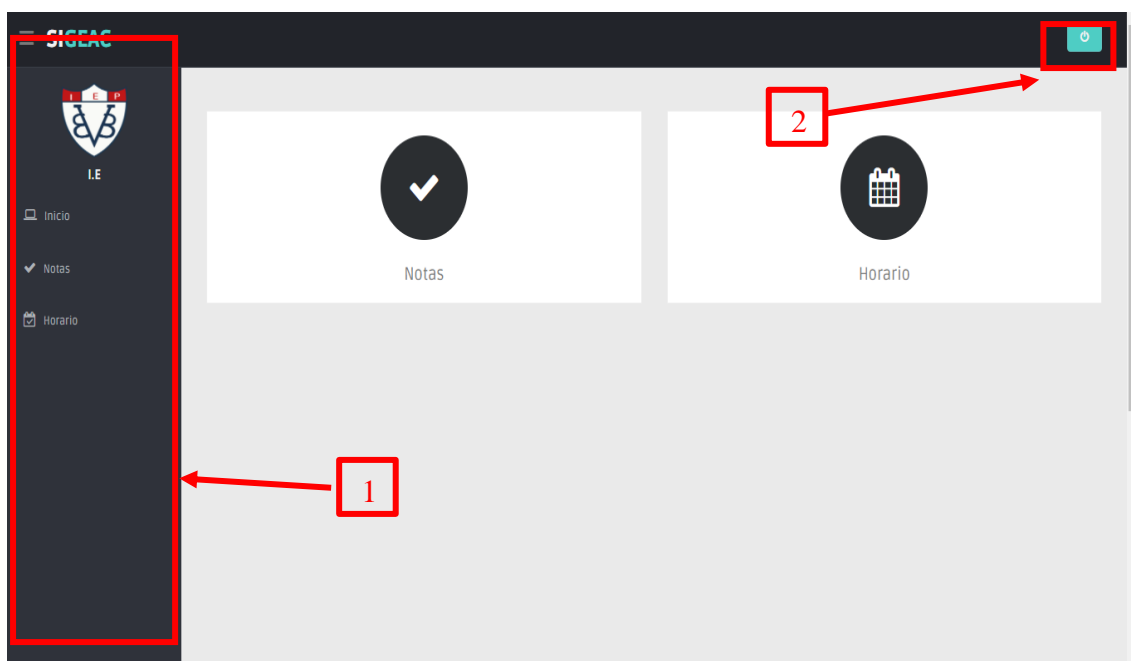


Figura 37. Menú Estudiante

- **Menú Perfil Estudiante**
- **Cerrar Sesión**

Permite al estudiante salir de la aplicación web de forma correcta y segura.

Para ello seleccionar con un clic izquierdo en el siguiente ícono:

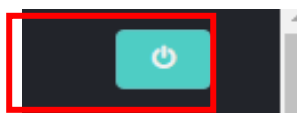


Figura 38. Botón Cerrar Sesión

Muestra el siguiente anuncio y nuevamente dará un clic izquierdo en Cerrar para salir de manera segura de la aplicación web Institucional.

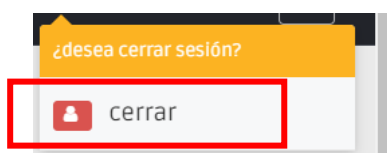


Figura 39. Cerrar Sesión

- **Menú Académico**
- **Inicio**

Se mostrará en la parte derecha de la pantalla dos botones de acceso:

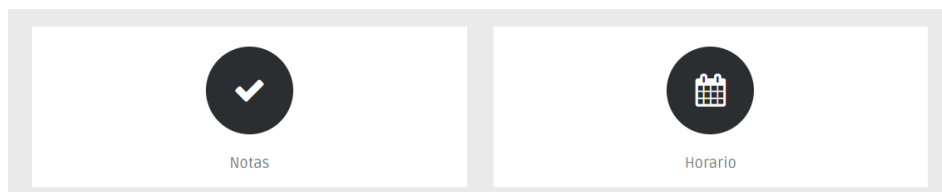


Figura 40. Botones de acceso

- **Notas**

Muestra al estudiante todos los cursos que lleva durante el año escolar junto con el resultado de las notas obtenidas según las evaluaciones correspondientes.

> Registro de notas						
CURSO	PF BI	PF BII	PF BIII	PF IV	PMD BIM..	DETA.
Matemática	20	15	0	0	9	
Comunicación	20	17	0	0	9	
Personal social	20	17	0	0	9	
Arte y cultura	18	16	0	0	8	
Ciencia y tecnología	18	15	0	0	8	
Educación física	18	10	0	0	7	
Educación religiosa	20	20	0	0	10	

Figura 41. Tabla de Notas

- **Detalle de notas**

Muestra todas las notas obtenidas durante todos los bimestres académicos. para conocer a detalle las notas damos clic izquierdo en el icono de color verde del lado derecho de la pantalla del curso de preferencia.

> Registro de notas						
CURSO	PF BI	PF BII	PF BIII	PF IV	PMD BIM..	DETA.
Matemática	20	15	0	0	9	
Comunicación	20	17	0	0	9	
Personal social	20	17	0	0	9	
Arte y cultura	18	16	0	0	8	
Ciencia y tecnología	18	15	0	0	8	
Educación física	18	10	0	0	7	
Educación religiosa	20	20	0	0	10	

Figura 42. Reporte de Notas

A continuación, aparecerá la ventana con los detalles de notas según el curso .

Notas																					
EV1 BI	EV2 BI	EV3 BI	EF BI	PF BI	EV1 BII	EV2 BII	EV3 BII	EF BII	PF BII	EV1 BIII	EV2 BIII	EV3 BIII	EF BIII	PF BIII	EV1 BIV	EV2 BIV	EV3 BIV	EF BIV	PF BIV	PMD BIM.	
20	20	20	20	20	10	10	20	20	15	0	0	0	0	0	0	0	0	0	0	0	9

Cancelar

Figura 43. Detalles de Notas

Para salir, dar clic en cancelar y nos regresará a la página anterior.

▪ Horario

Muestra la programación de horarios de clases, según: día, hora, curso, docente, tipo de ambiente.

Ver Horario									
Lunes	Viernes	Lunes	Jueves	Martes	Martes	Miércoles	Viernes	Miércoles	Jueves
08:00 10:30	08:00 10:30	11:00 13:00	11:00 13:00	08:00 10:30	11:30 13:00	08:00 10:30	11:00 13:00	11:00 13:00	08:00 10:30
Matemática Arenas Bustamante Amelia Veronica	Matemática Arenas Bustamante Amelia Veronica	Comunicación Arenas Bustamante Amelia Veronica	Comunicación Arenas Bustamante Amelia Veronica	Personal social Arenas Bustamante Amelia Veronica	Arte y cultura Arenas Bustamante Amelia Veronica	Ciencia y tecnología Arenas Bustamante Amelia Veronica	Ciencia y tecnología Arenas Bustamante Amelia Veronica	Educación física Arenas Bustamante Amelia Veronica	Educación religiosa Arenas Bustamante Amelia Veronica
						Aula 01	Aula 01	Aula 01	Aula 01

Figura 44. Programación de Horario

Modo de Acceso: Perfil Docente

Abrir su navegador de preferencia e ingrese la siguiente dirección URL:

http://localhost/Sis-VictoriaBarciaBoniffati/view_est/VLogin_Usuario_apoderado.php

Después ingresar su Usuario y Contraseña ,

Es de carácter obligatorio el llenado de ambos campos de Usuario y contraseña. Luego daremos clic en Iniciar Sesión y tendremos acceso a la aplicación web. (en caso de no completar ambos campos no tendrá acceso).

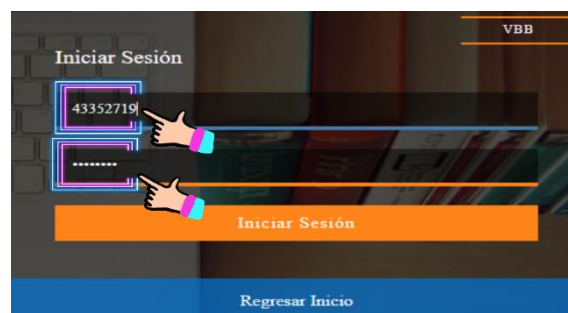


Figura 45. Inicio al perfil de docente

Ejemplo:

Usuario: DNI del Apoderado.

Contraseña: Ingresar la contraseña otorgada por el administrador de la I.E.

Iniciar Sesión: Proceso de ingreso al menú principal de la aplicación web modo estudiante.

Funcionalidad General

La interfaz de la aplicación web se divide en dos zonas:

- **Menú Perfil Apoderado**
- **Menú Académico**

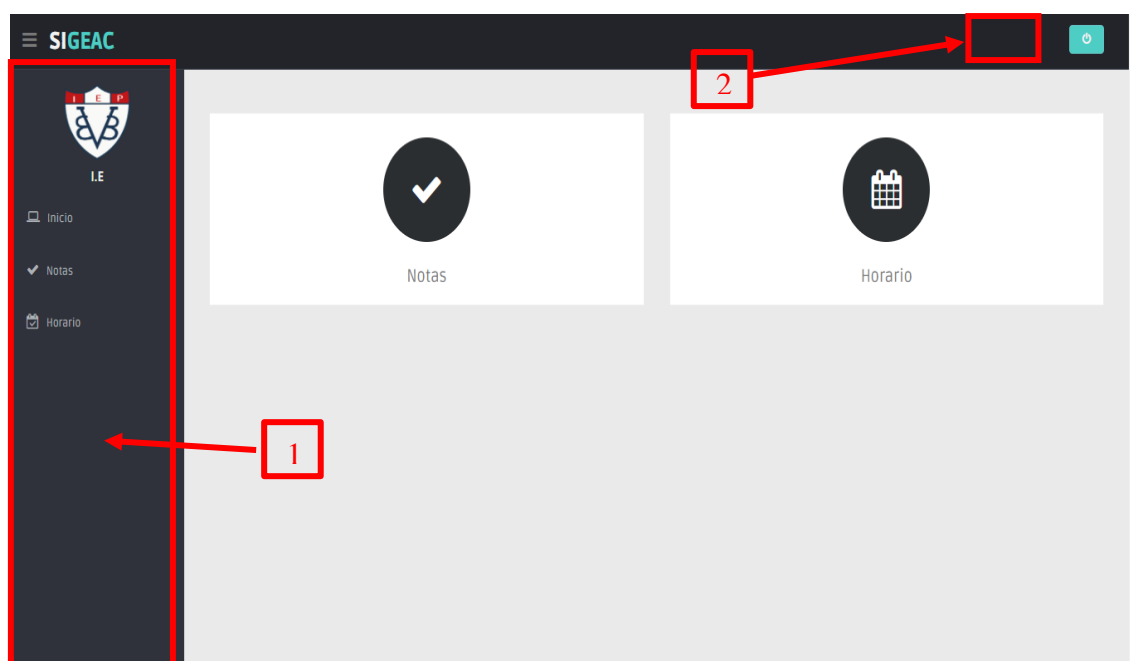


Figura 46. Página Principal Estudiante

- **Menú Perfil Apoderado**
- **Cerrar Sesión**

Permite al apoderado salir de la aplicación web de forma correcta y segura.

Para ello seleccionar con un clic izquierdo en el siguiente ícono:

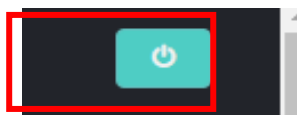


Figura 47. Botón de Cierre de Sesión

Muestra el siguiente anuncio y nuevamente dará un clic izquierdo en “Cerrar” para salir de manera segura de la aplicación web Institucional.

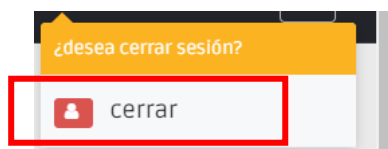


Figura 48. Cerrar Sesión

- **Menú Académico**
- **2.1. Inicio**

Se mostrará en la parte derecha de la pantalla dos botones de acceso:



Figura 49. Botones de acceso

- **Notas**

Muestra al apoderado todos los cursos que lleva sus hijos durante el año escolar junto con el resultado de las notas obtenidas según las evaluaciones del docente por curso.

 A screenshot of a mobile application interface showing a table titled "> Registro de notas". The table has columns for "CURSO", "PF BI", "PF BII", "PF BIII", "PF IV", "PMD BIM..", and "DETA.". The table is highlighted with a red border.

CURSO	PF BI	PF BII	PF BIII	PF IV	PMD BIM..	DETA.
Matemática	20	15	0	0	9	[Icono]
Comunicación	20	17	0	0	9	[Icono]
Personal social	20	17	0	0	9	[Icono]
Arte y cultura	18	16	0	0	8	[Icono]
Ciencia y tecnología	18	15	0	0	8	[Icono]
Educación física	18	10	0	0	7	[Icono]
Educación religiosa	20	20	0	0	10	[Icono]

Figura 50. Reporte de Notas

▪ Detalle de Notas

Muestra todas las notas obtenidas durante todos los bimestres académicos.

para conocer a detalle las notas damos clic izquierdo en el icono color verde del lado derecho de la pantalla del curso de preferencia.

> Registro de notas						
CURSO	PF BI	PF BII	PF BIII	PF IV	PMD BIM..	DETA.
Matemática	20	15	0	0	9	
Comunicación	20	17	0	0	9	
Personal social	20	17	0	0	9	
Arte y cultura	18	16	0	0	8	
Ciencia y tecnología	18	15	0	0	8	
Educación física	18	10	0	0	7	
Educación religiosa	20	20	0	0	10	

Figura 51. Registro de Notas

A continuación, aparecerá la ventana con los detalles de notas según el curso .

Notas																					
EV1 BI	EV2 BI	EV3 BI	EF BI	PF BI	EV1 BII	EV2 BII	EV3 BII	EF BII	PF BII	EV1 BIII	EV2 BIII	EV3 BIII	EF BIII	PF BIII	EV1 BIV	EV2 BIV	EV3 BIV	EF BIV	PF BIV	PMD BIM.	
20	20	20	20	20	10	10	20	20	15	0	0	0	0	0	0	0	0	0	0	0	9

Cancelar

Figura 52. Detalles de Notas

Para salir, dar clic en “cancelar” y nos regresará a la página anterior.

▪ Horario

Pone en conocimiento al apoderado la programación de horarios de clases, según: día, hora, curso, docente, tipo de ambiente. De esta manera el apoderado conocerá de manera eficiente la localización de su menor hijo.

Lunes	Viernes	Lunes	Jueves	Martes	Martes	Miércoles	Viernes	Miércoles	Jueves
08:00 10:30	08:00 10:30	11:00 13:00	11:00 13:00	08:00 10:30	11:30 13:00	08:00 10:30	11:00 13:00	11:00 13:00	08:00 10:30
Matemática	Matemática	Comunicación	Comunicación	Personal social	Arte y cultura	Ciencia y tecnología	Ciencia y tecnología	Educación física	Educación religiosa
Arenas Bustamante Amelia Veronica	Arenas Bustamante Amelia Veronica	Arenas Bustamante Amelia Veronica	Arenas Bustamante Amelia Veronica	Arenas Bustamante Amelia Veronica	Arenas Bustamante Amelia Veronica	Arenas Bustamante Amelia Veronica	Arenas Bustamante Amelia Veronica	Arenas Bustamante Amelia Veronica	Arenas Bustamante Amelia Veronica
Aula 01	Aula 01	Aula 01	Aula 01	Aula 01	Aula 01	Aula 01	Aula 01	Aula 01	Aula 01

Figura 53. Reporte de Horario

3.5 Limitaciones

- Falta de respuesta al momento de solicitar información de la institución educativa.
- Coordinación no fluida entre el personal administrativo debido a la solicitud de fotos para el diseño de la pagina del aplicativo web.
- Se encontraron dificultades en los documentos que nos brindaron de referencias.
- El factor tiempo fue in inconveniente, ya que se cruzaba el trabajo laboral del los integrantes del proyecto.
- Se observó que la institución educativa tenía problemas de pérdidas de documentos y registros
- Tuvimos dificultad para entrevistar a los docentes, alumnos, padres de familia o apoderado para recolectar su información ya que algunos alumnos se ausentaban, padres de familias que no recogerían a sus hijos y se demostró que los docentes no se encontraban disponibles en el momento de nuestra visita a esta casa de estudios.

CAPITULO IV

RESULTADOS

RESULTADOS

Se trabajo a través de esta aplicación Web , en la cual se apreció una mejora para la gestión académica de la Institución Educativa “Victoria Barcia Boniffatti; es decir, el tiempo de las tareas administrativas, académicas y la reducción de uso de los documentos físicos ya que no se dañarán, no se extraviarán ni serán manipuladas por personas con finalidad de dañar a la institución.

Esta aplicación ayudará a los procesos administrativos para tener un mejor control de las notas, asistencias y promoción de años escolares y saber cómo se encuentra la gestión académica. Por otro lado, gracias a este sistema el área administrativa tendrá un control de toda la información en línea de la parte académica.

La ventaja de la aplicación es que el usuario (estudiantes, profesores, personal externo a la institución) podrá acceder a la plataforma y enviar su trámite las 24 horas del día.

En esta etapa fue necesario determinar la influencia de la implementación de una aplicación web, la cual mejorará el proceso de gestión académica de la institución educativa Victoria Barcia Boniffatti, con la finalidad de satisfacer a todos los usuarios que serán beneficiados con esta aplicación.

CAPITULO V

CONCLUSIONES Y RECOMENDACIONE

CONCLUSIONES

- a) El uso de la aplicación web de gestión académica permitirá que la institución educativa Victoria Barcia Boniffatti optimice los procesos que se encuentran dentro de la gestión académica.
- b) Los padres o apoderados podrán ver los reporte de los registros de notas, asistencia rápidamente sin volumen de hojas impresas, ayudando así a reducir el tiempo de espera y evitando el exceso de hojas impresas.
- c) El personal administrativo y docente tendrán el manejo mas eficiente en el control de los registros de matrícula, asistencias y actas de notas.
- d) Los alumnos tendrán sus resultados en menos tiempo gracias a la implementación de la aplicación web que ayudara a mejorar el proceso de gestión académica de la institución educativa Victoria Barcia Boniffatti

RECOMENDACIONES

- a) La institución educativa debe contar con un personal administrativo con las capacidades de manejo de la aplicación web.
- b) La institución debe contar con un equipo de cómputo en cada aula y laboratorio con una alta red de calidad para su buen uso de la aplicación.
- c) El administrador debe realizar un mantenimiento a los servidores y mantener actualizado la aplicación web para un óptimo rendimiento en el proceso de gestión académica.
- d) El administrador no debe realizar un mal uso de la aplicación web ya que se encuentra muchos registros con información de la gestión académica de la institución.

REFERENCIAS

- Aguirre, R. (2020, 2 de julio). *Crear una imagen con PHP*. Altruistas Org. Revista digital de cultura, ocio y diversión. <https://altruistas.org/crear-una-imagen-con-php/>
- Arias, Á. (2014). *Aprende a Programar con Ajax*. IT Campus Academy Editorial.
- Intituto Fomento Ocupacional, FOC. (2016, 28 de junio). *El motor de la tecnología AJAX*. <https://acortar.link/c1Jvz>
- Ayauja, E. y Delgado, M. D. (2017). *Estudio y Diseño de una Aplicación informática para el control de Matrícula de La Facultad de Agronomía de la Universidad Nacional San Luis Gonzaga de Ica*. [Tesis de pregrado, Universidad Nacional San Luis Gonzaga de Ica]. Archivo Digital. <http://repositorio.unica.edu.pe/handle/UNICA/2865>
- Bahit, E. (2011). *POO y MVC en PHP*. Editorial Creative Commons.
- Bernardo, K., y Paredes J. (2016). *Aplicación de la metodología Six Sigma para mejorar el proceso de registro de matrícula, en la Universidad Autónoma del Perú*. [Tesis de pregrado, Universidad Autónoma del Perú]. Archivo digital. <http://repositorio.autonoma.edu.pe/handle/AUTONOMA/339>
- Blancarte, O. (2014, 21 de julio). Patrón de diseño Modelo Vista Controlador (MVC). <https://www.oscarblancarteblog.com/2014/07/21/patron-de-diseno-modelo-vista-controlador-mvc/>
- Burbank, D., y Hoberman, S. (2011). *Data modeling made simple with CA ERwin Data Modeler R8*. Technics Publications. <https://www.dataversity.net/erwin-data-modeling-named-top-solution-dbta-readers/#>

- Colimba, I. P. (2016). *Desarrollo de un portal web utilizando el Framework Laravel y Bootstrap para la difusión turística de la Hostería Fabricios en la parroquia de Ambuquí*. [Tesis de pregrado, Universidad Técnica del Norte]. Archivo digital. <http://repositorio.utn.edu.ec/handle/123456789/5698>
- Cumbo, A. (2016, 2 de agosto). *Los Mejores 26 Plugins JQuery que tienes que conocer*. <https://cumboandrea.me/26-jquery-plugins-muy-utiles/>
- Flanagan, D.(2007). JavaScript . AMAYA O'RAILLY Editorial.
- Gómez, E. (2015, 13 de agosto). *De MySQL a Arduino, leer datos de la base de datos*. rinconingenieril. <https://www.rinconingenieril.es/de-mysql-arduino/#comments>
- Granado, M. (2009). *PHP5*. Editorial Grupo ANAYA S.A.
- Hernández E. A. (2020). *Desarrollo de una aplicación web con el Framework Bootstrap y el precompilador Sass para la gestión de pedidos de productos agrícolas de la Empresa El Chagra*. [Tesis de pregrado, Escuela Superior Politécnica de Chimborazo]. Archivo digital. <http://dspace.esPOCH.edu.ec/handle/123456789/13654>
- Huaman, J. S. (2019). *Influencia de una Aplicación móvil en el proceso de Matrícula y Consulta académica en los estudiantes de la facultad de ingeniería de la Universidad Privada Antonio Guillermo Urrelo*. [Tesis de pregrado, Universidad Nacional de Cajamarca], Archivo Digital. <http://repositorio.unc.edu.pe/handle/UNC/3249>
- Huertas, A. (2019, 07 de febrero). *Sistemas Operativos en Red*. Angelhuertasdam. <http://angelhuertasdam.blogspot.com/2019/02/sistemas-operativos-en-red.html>
- Luján-Mora, S. (2002). *Programación de aplicaciones web: historia, principios básicos y clientes web*. Editorial Club Universitario.

- Montoya, C. (2019, 20 de febrero). *Hablemos de javascript*. Medium.
<https://acortar.link/E7Dzf>
- Marquez, J. A. (2018). *Implementacion de un sistema informatico de matrículas y pensiones de la I.E.P. Santa Isabel de Hungria – Casma* [Tesis de pregrado, Universidad Catolica los Angeles de Chimbote], Archivo digital.
<http://repositorio.uladech.edu.pe/handle/123456789/9227>
- Noguera, L. (2018, 20 de abril). *Logros de la Web 2.0*
<https://medium.com/@larisanoguera8/logros-de-la-web-2-0-908623f8d297>
- Ramos, A. y Ramos, M. J. (2014). *Aplicaciones Web 2.0* . Editorial PARANINFO.
- Spona, H. (2010). Programación de bases de datos con MYSQL y PHP. Marcombo.
- Uribe, A. R. (2017). *Propuesta del sistema web para la gestión de Matrícula y Registro de Notas del nivel Secundario del Colegio Privado Peruano Americano – Huaraz* [Tesis de Titulación, Universidad Catolica los Angeles de Chimbote]. Archivo Digital <http://repositorio.uladech.edu.pe/handle/123456789/13059>
- Zeas, N. (2015). *Desarrollo de una Plataforma Web para una Red de Monitorización de la Calidad del Aire*. [Tesis en Ingeniería Informática. Universidad Politécnica de Cataluña]. Archivo digital.
<https://upcommons.upc.edu/handle/2099.1/24936>

APÉNDICES

Apéndice B. Cronograma de Presupuesto

Materiales			
Detalle	Cantidad	Valor (s/.)	Total(s/.)
Laptop	3	1500	S/. 4,500.00
Impresora	1	650	S/. 650.00
Papel Bond	3	22	S/. 66.00
Memoria USB	3	50	S/. 150.00
Tinta de Impresión	3	50	S/. 150.00
Sub Total			S/. 5,516.00
Servicios			
Detalle	Cantidad	Valor (s/.)	Total(s/.)
Internet	1	250	S/. 250.00
Comunicaciones	1	200	S/. 200.00
Movilidad	1	450	S/. 450.00
Copias	100	0.1	S/. 10.00
Otros	1	300	S/. 300.00
Sub Total			S/. 1,210.00
Total General			S/. 6,726.00

Apéndice C. Manual de Usuario

Manual de Usuario - Aplicación web Gestión Académica SIGEAC

Para ingresar a la aplicación web a realizar cualquier actividad o consultas, el usuario deberá ingresar su “Usuario” y “Contraseña” , existen cuatro modos de acceso según sea el caso:

- e. Administrador
- f. Docente
- g. Apoderado
- h. Estudiante

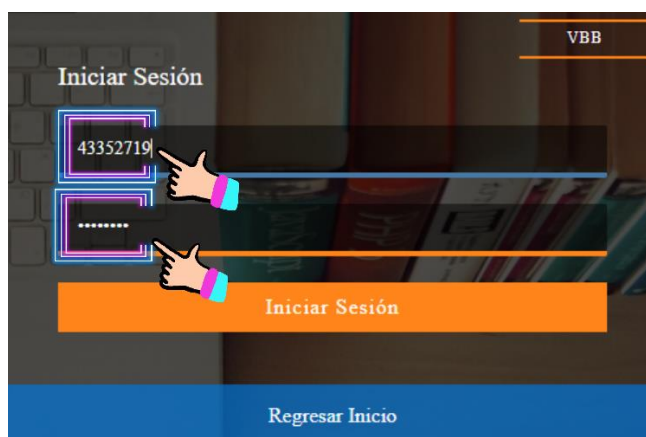
Para cualquier de los cuatro casos; será de carácter obligatorio rellenar según corresponda los dos campos de Usuario y contraseña. Luego daremos clic en Iniciar Sesión y tendremos acceso a la aplicación web. (en caso de no completar ambos campos no tendrá acceso).

Perfil: Administrador/secretaria

Abrir el navegador y colocar la siguiente dirección URL:

http://localhost/SisVictoriaBarciaBoniffati/view/VLogin_usuario_personal_administrativo.php

A continuación, ingresar su usuario y contraseña, después clic en iniciar sesión.



MATRÍCULA

El personal encargado del área de secretaría académica se encargará de **registrar la información del nuevo estudiante** y apoderado; en caso de **que el estudiante es antiguo** y pasa de grado académico solamente se realizara la **Actualización** correspondiente.

A continuación, se muestra un ejemplo para ambos casos .

Nueva Matrícula

Realizaremos estos tres simples pasos para abrir la ventana de nueva matrícula:

1. Pulse la opción para acceder a la Matricula.
2. Pulse la opción para acceder a la Asistencia
3. Pulse la opción para acceder a las Notas del estudiante.
4. Pulse la opción para asignar horario.
5. Pulse la opción para registrar los docentes.
6. Pulse la opción para registrar a los estudiantes.
7. Mensajes recientes.
8. Cerrar sesión.

Registrar matrícula

Año: 2020

Estudiante: Seleccionar estudiante

Apoderado: Alvarez Ramos Diego Jose

Responsable matricula: Fernandez Maldonado Juan Carl

Grado \ Nivel \ Sección: Grado ~ Nivel ~ Sec

Fecha de matricula: 08-12-2020

Estado: Matriculado

Cancelar Guardar

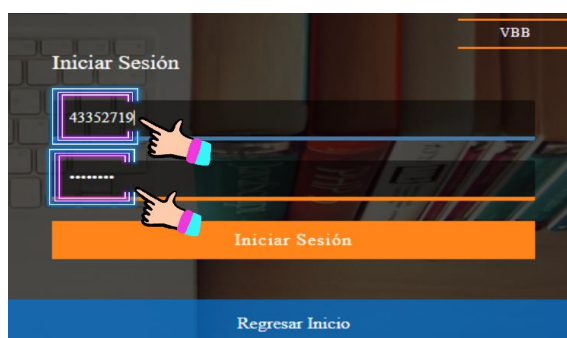
Todos los campos serán rellenos de manera obligatorio, una vez completados se le dará el estado de “Matriculado” y clic en Guardar. Automáticamente todos los datos ingresados serán almacenados en la base de datos de la Institución educativa. En caso de querer finalizar la inscripción por algún motivo o error solo daremos clic en cancelar.

Perfil: Docente

Abrir el navegador y colocar la siguiente dirección URL:

http://localhost/Sis-VictoriaBarciaBoniffati/view_doc/VLogin_usuario_docente.php

A continuación, ingresar su usuario y contraseña, después clic en iniciar sesión.

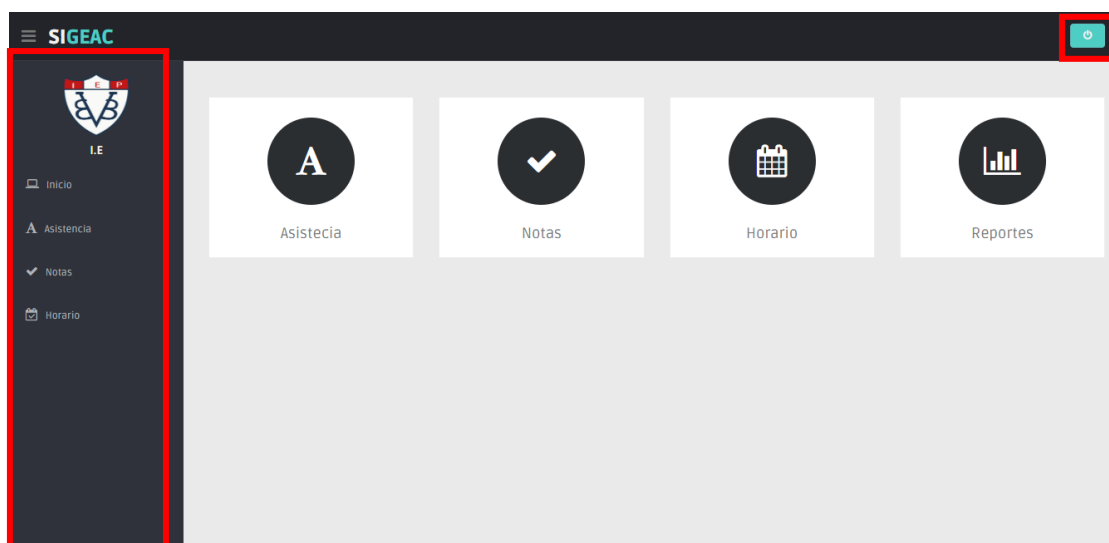


FUNCIONALIDAD GENERAL

La interfaz de la aplicación web se divide en dos zonas:

- **Menú Perfil Docente**

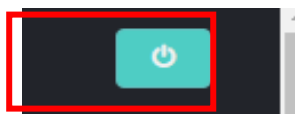
- **Menú AcadémicoMenú Perfil Docente**



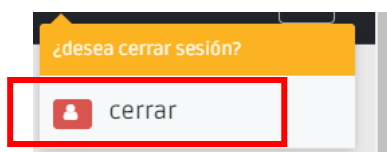
- **Cerrar Sesión**

Permite al Docente salir de la aplicación web de forma correcta y segura.

Para ello seleccionar con un clic izquierdo en el siguiente ícono:



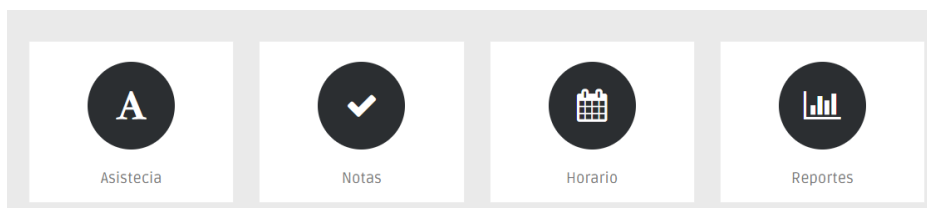
Muestra el siguiente anuncio y nuevamente dará un clic izquierdo en Cerrar para salir de manera segura de la aplicación web Institucional.



Menú Académico

- **Inicio**

Se mostrará en la parte derecha de la pantalla los siguientes botones de acceso:



Asistencia

Muestra al docente la relación de estudiantes según el grado y el curso.

Asistencia

1ro - Primaria - A Matemática - Primaria 11-12-2020

Grado	Nivel	Sección	Curso	Fecha	Estudiante	A	T	J	J	E	Edi.
1ro	Primaria	A	Matemática	11-12-2020	Alvarez Medrano Carla Melisa	si	no	no	no		<input type="checkbox"/>
1ro	Primaria	A	Matemática	11-12-2020	Aguilar Castillo Julio Cesar	si	si	no	no		<input type="checkbox"/>
1ro	Primaria	A	Matemática	11-12-2020	Alvarado Castro Maria Juana	si	no	no	no		<input type="checkbox"/>
1ro	Primaria	A	Matemática	11-12-2020	Arias Juarez Miguel Freddy	no	no	si	si		<input type="checkbox"/>

Después de pasar lista a la hora indicada el docente dará clic en “Guardar”.

Notas

En esta opción el docente podrá ingresar las notas obtenidas de los estudiantes según el grado y curso.

REGISTRAR NOTAS ACTUALIZAR NOTAS

REGISTRAR NOTAS Grado - Nivel - Sec

Estudiante	EVI BI	EVI BII	EVI BIII	PP BII	EVI BIII	EVI BIII	PP BIII	EVI IV	EVI IV	EVI IV	PP IV
Docente Arenas Bustamante Amelia Veronica											
Curso Matemática - Primaria											

+ Guardar

Al finalizar el llenado de notas el docente deberá dar clic en “Guardar” para que todas las modificaciones queden grabadas correctamente.

Horario

Facilita al docente la programación de horarios de clases, según: día, hora, curso, tipo de ambiente.

Lunes	Viernes	Lunes	Jueves	Martes	Martes	Miércoles	Viernes	Miércoles	Jueves
08:00 10:30	08:00 10:30	11:00 13:00	11:00 13:00	08:00 10:30	11:30 13:00	08:00 10:30	11:00 13:00	11:00 13:00	08:00 10:30
Matemática 1ro Primaria A Aula 01	Matemática 1ro Primaria A Aula 01	Comunicación 1ro Primaria A Aula 01	Comunicación 1ro Primaria A Aula 01	Personal social 1ro Primaria A Aula 01	Arte y cultura 1ro Primaria A Aula 01	Ciencia y tecnología 1ro Primaria A Aula 01	Ciencia y tecnología 1ro Primaria A Aula 01	Educación física 1ro Primaria A Aula 01	Educación religiosa 1ro Primaria A Aula 01

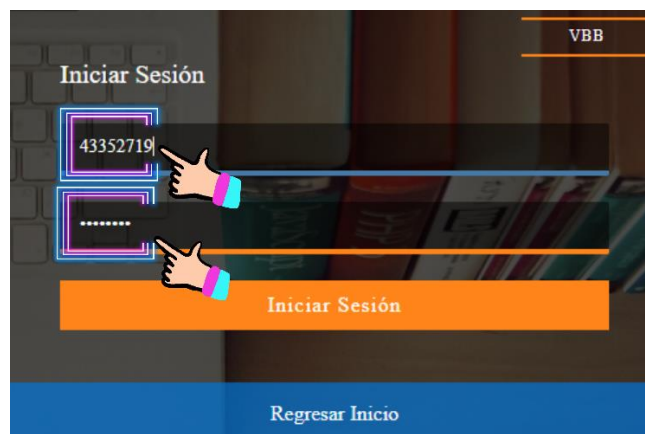
Perfil Estudiante

Abrir su navegador de preferencia e ingrese la siguiente dirección URL:

http://localhost/Sis-VictoriaBarciaBoniffati/view_est/VLogin_usuario_estudiante.php

Después ingresar su Usuario y Contraseña ,

Es de carácter obligatorio el llenado de ambos campos de Usuario y contraseña. Luego daremos clic en Iniciar Sesión y tendremos acceso a la aplicación web. (en caso de no completar ambos campos no tendrá acceso).



Ejemplo:

Usuario: DNI del estudiante.

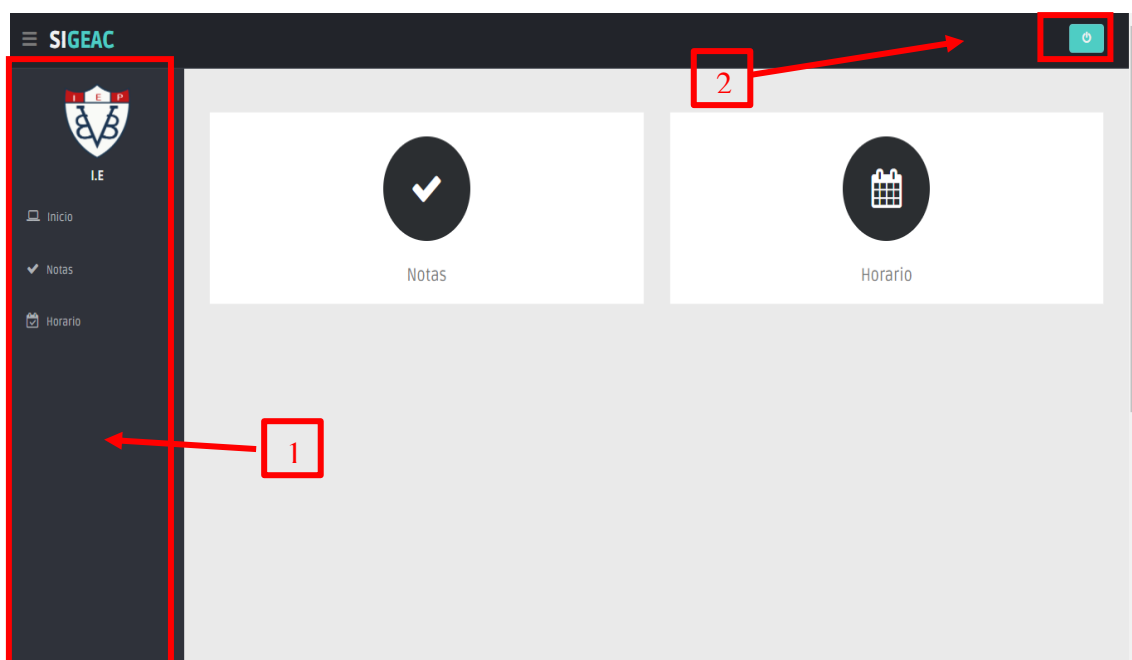
Contraseña: Ingresar la contraseña otorgada por el administrador de la I.E.

Iniciar Sesión: Proceso de ingreso al menú principal de la aplicación web modo estudiante.

FUNCIONALIDAD GENERAL

La interfaz de la aplicación web se divide en dos zonas:

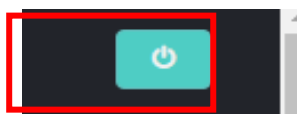
- **Menú Perfil Estudiante**
- **Menú Académico**



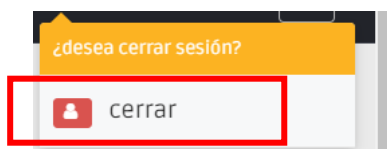
- **Menú Perfil Estudiante**
- **Cerrar Sesión**

Permite al estudiante salir de la aplicación web de forma correcta y segura.

Para ello seleccionar con un clic izquierdo en el siguiente ícono:



Muestra el siguiente anuncio y nuevamente dará un clic izquierdo en Cerrar para salir de manera segura de la aplicación web Institucional.



- **Menú Académico**

- **Inicio**

Se mostrará en la parte derecha de la pantalla dos botones de acceso:



- **Notas**

Muestra al estudiante todos los cursos que lleva durante el año escolar junto con el resultado de las notas obtenidas según las evaluaciones correspondientes.

> Registro de notas						
CURSO	PF BI	PF BII	PF BIII	PF IV	PMD BIM..	DETA.
Matemática	20	15	0	0	9	
Comunicación	20	17	0	0	9	
Personal social	20	17	0	0	9	
Arte y cultura	18	16	0	0	8	
Ciencia y tecnología	18	15	0	0	8	
Educación física	18	10	0	0	7	
Educación religiosa	20	20	0	0	10	

- **Detalle de Notas**

Muestra todas las notas obtenidas durante todos los bimestres académicos.

para conocer a detalle las notas damos clic izquierdo en el icono color verde del lado derecho de la pantalla del curso de preferencia.

> Registro de notas

CURSO	PF BI	PF BII	PF BIII	PF BIV	PMD BIM..	DETA.
Matemática	20	15	0	0	9	
Comunicación	20	17	0	0	9	
Personal social	20	17	0	0	9	
Arte y cultura	18	16	0	0	8	
Ciencia y tecnología	18	15	0	0	8	
Educación física	18	10	0	0	7	
Educación religiosa	20	20	0	0	10	

A continuación, aparecerá la ventana con los detalles de notas según el curso .

Notas

EV1 BI	EV2 BI	EV3 BI	EF BI	PF BI	EV1 BII	EV2 BII	EV3 BII	EF BII	PF BII	EV1 BIII	EV2 BIII	EV3 BIII	EF BIII	PF BIII	EV1 BIV	EV2 BIV	EV3 BIV	EF BIV	PF BIV	PMD BIM.	
20	20	20	20	20	10	10	20	20	15	0	0	0	0	0	0	0	0	0	0	0	9

Cancelar

Para salir, dar clic en cancelar y nos regresará a la página anterior.

▪ Horario

Muestra la programación de horarios de clases, según: día, hora, curso, docente, tipo de ambiente.

Ver Horario

Lunes	Viernes	Lunes	Jueves	Martes	Martes	Miércoles	Viernes	Miércoles	Jueves
08:00 10:30	08:00 10:30	11:00 13:00	11:00 13:00	08:00 10:30	11:30 13:00	08:00 10:30	11:00 13:00	11:00 13:00	08:00 10:30
Matemática Arenas Bustamante Amelia Verónica	Matemática Arenas Bustamante Amelia Verónica	Comunicación Arenas Bustamante Amelia Verónica	Comunicación Arenas Bustamante Amelia Verónica	Personal social Arenas Bustamante Amelia Verónica	Arte y cultura Arenas Bustamante Amelia Verónica	Ciencia y tecnología Arenas Bustamante Amelia Verónica	Ciencia y tecnología Arenas Bustamante Amelia Verónica	Educación física Arenas Bustamante Amelia Verónica	Educación religiosa Arenas Bustamante Amelia Verónica
Aula 01	Aula 01	Aula 01	Aula 01	Aula 01	Aula 01	Aula 01	Aula 01	Aula 01	Aula 01

Modo de Acceso: Perfil Docente

Abrir su navegador de preferencia e ingrese la siguiente dirección URL:

http://localhost/Sis-VictoriaBarciaBoniffati/view_est/VLogin_Usuario_apoderado.php

Después ingresar su Usuario y Contraseña ,

Es de carácter obligatorio el llenado de ambos campos de Usuario y contraseña. Luego daremos clic en Iniciar Sesión y tendremos acceso a la aplicación web. (en caso de no completar ambos campos no tendrá acceso).



Ejemplo:

Usuario: DNI del Apoderado.

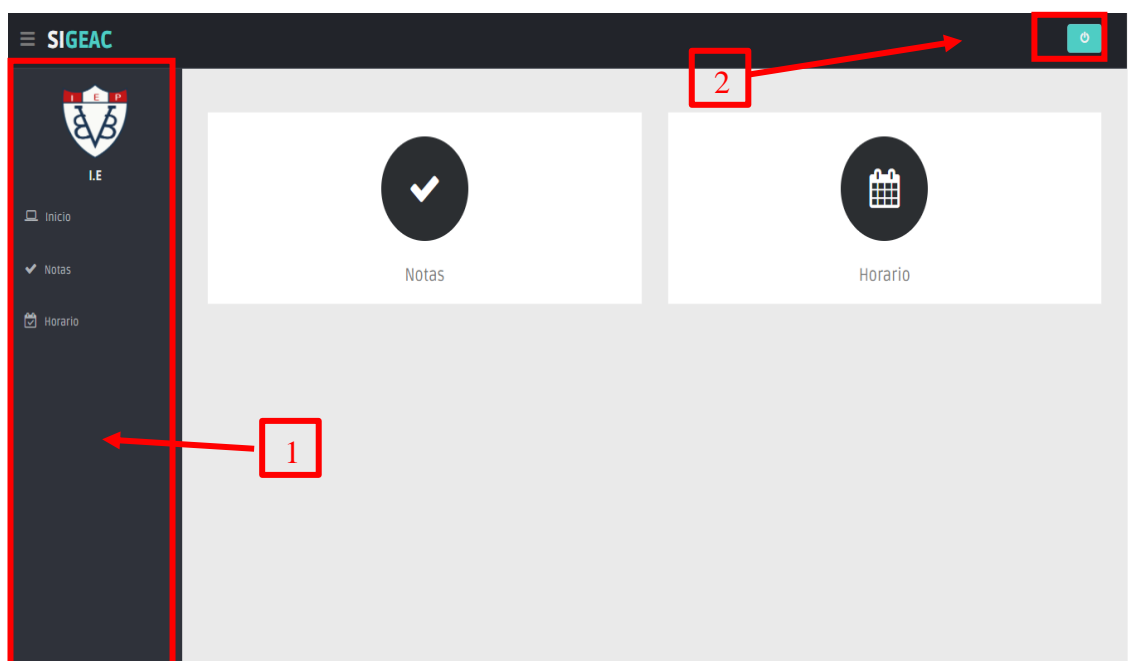
Contraseña: Ingresar la contraseña otorgada por el administrador de la I.E.

Iniciar Sesión: Proceso de ingreso al menú principal de la aplicación web modo estudiante.

FUNCIONALIDAD GENERAL

La interfaz de la aplicación web se divide en dos zonas:

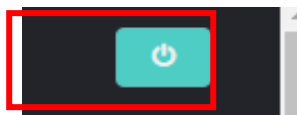
- **Menú Perfil Apoderado**
- **Menú Académico**



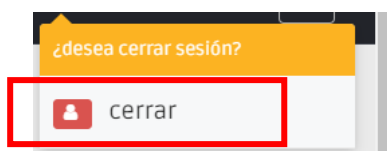
- **Menú Perfil Apoderado**
- **Cerrar Sesión**

Permite al apoderado salir de la aplicación web de forma correcta y segura.

Para ello seleccionar con un clic izquierdo en el siguiente ícono:



Muestra el siguiente anuncio y nuevamente dará un clic izquierdo en “Cerrar” para salir de manera segura de la aplicación web Institucional.



- **Menú Académico**
- **2.1. Inicio**

Se mostrará en la parte derecha de la pantalla dos botones de acceso:



- **Notas**

Muestra al apoderado todos los cursos que lleva sus hijos durante el año escolar junto con el resultado de las notas obtenidas según las evaluaciones del docente por curso.

> Registro de notas						
CURSO	PF BI	PF BII	PF BIII	PF IV	PMD BIM..	DET.
Matemática	20	15	0	0	9	
Comunicación	20	17	0	0	9	
Personal social	20	17	0	0	9	
Arte y cultura	18	16	0	0	8	
Ciencia y tecnología	18	15	0	0	8	
Educación física	18	10	0	0	7	
Educación religiosa	20	20	0	0	10	

