



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

**FACULTAD DE INGENIERIA
CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES**

TÍTULO:

**DESARROLLO E IMPLEMENTACIÓN DE UN SISTEMA WEB
GESTOR DE HORARIOS ACADÉMICOS AUTOMÁTICOS
PARA LA UNIDAD EDUCATIVA FEDERICO GONZÁLEZ
SUAREZ DEL CANTÓN DURÁN**

AUTORES:

Borja Gutiérrez, Petita María; Triviño Calderón, Jean Carlos

**Trabajo de Titulación previo a la Obtención del Título de:
INGENIERO EN SISTEMAS COMPUTACIONALES**

TUTOR:

Ing. Sánchez Calle, Roberto Eduardo, Mgs.

**Guayaquil, Ecuador
23 de Septiembre del 2016**



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

**FACULTAD DE INGENIERIA
CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES**

CERTIFICACIÓN

Certificamos que el presente trabajo fue realizado en su totalidad por **Borja Gutiérrez, Petita María y Triviño Calderón, Jean Carlos** como requerimiento parcial para la obtención del Título de **Ingeniero en Sistemas Computacionales**.

TUTOR

Ing. Sánchez Calle, Roberto Eduardo, Mgs.

DIRECTORA DE CARRERA

Ing. Guerrero Yépez, Beatriz Del Pilar Mgs.

Guayaquil, a los 23 días del mes de Septiembre del año 2016



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

**FACULTAD DE INGENIERIA
CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES**

DECLARACIÓN DE RESPONSABILIDAD

Nosotros,
Borja Gutiérrez, Petita María y Triviño Calderón, Jean Carlos

DECLARAMOS QUE:

El Trabajo de Titulación “**Desarrollo e Implementación de un Sistema Web Gestor de Horarios Académicos Automáticos para la Unidad Educativa Federico González Suarez del Cantón Durán**” previo a la obtención del Título de **Ingeniero en Sistemas Computacionales**, ha sido desarrollado en base a una investigación exhaustiva, respetando derechos intelectuales de terceros conforme las citas que constan al pie de las páginas correspondientes, cuyas fuentes se incorporan en la bibliografía. Consecuentemente este trabajo es de nuestra total autoría.

En virtud de esta declaración, me responsabilizo del contenido, veracidad y alcance científico del Trabajo de Titulación referido.

Guayaquil, a los 23 días del mes de Septiembre del año 2016

LOS AUTORES:

Borja Gutiérrez, Petita María

Triviño Calderón, Jean Carlos



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

**FACULTAD DE INGENIERIA
CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES**

AUTORIZACIÓN

Nosotros,
Borja Gutiérrez, Petita María y Triviño Calderón, Jean Carlos

Autorizamos a la Universidad Católica de Santiago de Guayaquil, la publicación en la biblioteca de la institución del Trabajo de Titulación: **“Desarrollo e Implementación de un Sistema Web Gestor de Horarios Académicos Automáticos para la Unidad Educativa Federico González Suarez del Cantón Durán”**, cuyo contenido, ideas y criterios son de nuestra exclusiva responsabilidad y total autoría.

Guayaquil, a los 23 días del mes de Septiembre del año 2016

LOS AUTORES:

Borja Gutiérrez, Petita María

Triviño Calderón, Jean Carlos

REPORTE URKUND



Documento [Trabajo Titulación Borja Triviño.docx](#) (D21481008)

Presentado 2016-08-23 22:02 (-05:00)

Recibido roberto.sanchez.ucsg@analysis.orkund.com

Mensaje [TTA16] [Mostrar el mensaje completo](#)

0% de esta aprox. 35 páginas de documentos largos se componen de texto presente en 0 fuentes.

AGRADECIMIENTO

Agradezco a Dios por darme la salud, sabiduría, fortaleza para poder lograr esta meta tan importante en mi vida y en mi carrera profesional, le agradezco por ayudarme en cada momento y estar ahí en todos y cada uno de los obstáculos que se me han presentado en la vida.

Agradezco a mis padres, José Luis y Juana María, que con su apoyo incondicional me han enseñado que con esfuerzo, trabajo y constancia, todo se puede alcanzar y nunca se debe dejar de luchar por lo que se desea. A mi mami Mariana y a mi tía Janeth por ayudarme siempre con todo lo que está a su alcance.

A mi amigo y compañero de tesis Jean Carlos quien me apoyó durante la elaboración de nuestro trabajo de titulación.

Borja Gutiérrez, Petita María

AGRADECIMIENTO

Agradezco principalmente a Dios por poner en mi camino a tan extraordinarias personas a lo largo de mi vida y en especial a mi familia que sin su apoyo incondicional esto no hubiera sido posible, en las buenas y malas siempre han estado cuando los necesito ayudándome a seguir adelante con mis estudios, dándome el mejor regalo que a un hijo se le puede dar y es una excelente educación académica.

Gracias a todos y cada uno de mis queridos y apreciados profesores, amig@s y compañer@s por guiarme, por cada experiencia vivida, por aportar con un granito en mi vida estudiantil y profesional que sin cada uno de ellos no tendría la preparación debida para afrontar este difícil y largo camino que ha sido.

Finalmente quiero agradecer enormemente a mi compañera tesis porque sin importar la complejidad que fue este gran reto pudimos terminar exitosamente este proyecto y al fin poder decir ¡Lo logramos Petita! Gracias por todas tus enseñanzas y por haber compartido conmigo estos meses de arduo trabajo y esfuerzo.

Triviño Calderón, Jean Carlos

DEDICATORIA

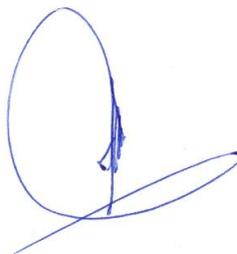
Hoy cumpla una etapa más en mi vida y quisiera dedicarle este proyecto de titulación a mi abuelito Rafael quien en mi vida me enseñó lo que significa superarse y que sin importar la situación en que me encuentre por más difícil que se vea, siempre hay que mantener ese positivismo y buen ánimo para seguir adelante y no rendirse nunca,

Triviño Calderón, Jean Carlos

Este trabajo de titulación se lo dedico a mi papá, José Luis Borja Plúas, quien ha sido, es y será siempre un pilar fundamental en mi vida. Es él quien con su ejemplo, me ha enseñado el significado de nunca rendirse, y siempre estuvo ahí ayudándome a levantar de mis caídas y dándome todos los ánimos necesarios para poder seguir adelante.

Borja Gutiérrez, Petita María

TRIBUNAL DE SUSTENTACION



Ing. Roberto Eduardo Sánchez Calle, Mgs.
PROFESOR TUTOR



Ing. Beatriz Del Pilar Guerrero Yépez, Mgs.
DIRECTORA DE CARRERA



Ing. César Adriano Salazar Tovar, Mgs.
COORDINADOR DE ÁREA



Ing. Ana Isabel Camacho Coronel, Mgs.
OPONENTE



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

**FACULTAD DE INGENIERIA
CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES**

CALIFICACIÓN

Ing. Roberto Eduardo Sánchez Calle, Mgs.
PROFESOR TUTOR

Ing. Beatriz Guerrero Yépez, Mgs.
DIRECTORA DE CARRERA

Ing. César Adriano Salazar Tovar, Mgs.
COORDINADOR DE ÁREA

Ing. Ana Isabel Camacho Coronel, Mgs.
OPONENTE

INDICE GENERAL

RESUMEN	xv
ABSTRACT	xvi
INTRODUCCIÓN.....	17
CAPITULO I: FUNDAMENTACIÓN CONCEPTUAL	18
1.1 Planteamiento del problema	18
1.2 Objetivo General	18
1.3 Objetivos Específicos	18
1.4 Alcance	19
1.4 Marco Teórico.....	20
CAPITULO II: METODOLOGIA DE LA INVESTIGACION.....	35
2.1 Tipo de Investigación.....	35
2.2 Enfoque Metodológico.....	35
2.3 Instrumento.....	36
2.4 Aplicación y Procesamiento: Herramientas estadísticas	37
2.5 Análisis de los resultados.....	38
CAPITULO III: PROPUESTA	46
3.1 Herramientas de desarrollo.....	46
3.1.1 Lenguaje programación.....	46
3.1.2 IDE	48
3.1.3 Frameworks	49
3.2 Base de Datos.....	50
3.3 Arquitectura de la solución.....	52
3.4 Seguridad	53
3.4.1 Seguridad a nivel de desarrollo.....	53
3.4.2 Seguridad en la base de datos.....	54
3.4.3 Seguridad en la web.....	54
3.5 Diagrama Entidad Relación.....	55
3.6 Casos de uso	57
3.7 Estudio de Factibilidad	68
3.7.1 Estudio de Factibilidad Técnica.....	68
3.7.2 Estudio de Factibilidad Económica.....	69
CAPITULO IV: CONCLUSIONES Y RECOMENDACIONES.....	70
4.1 Conclusiones.....	70

4.2 Recomendaciones	71
5. Bibliografía.....	72
6. Anexos	75
6.1 Anexo 1: Entrevista a Rectora de la Unidad Educativa Federico González Suarez del Cantón Durán.....	75
6.2 Anexo 2: Formato de encuesta a los Docentes de la Unidad Educativa Federico González Suarez del Cantón Durán.....	78
6.3 Anexo 3: Manual de Usuario.....	80
6.4 Anexo 4: Manual Técnico.....	115
AUTORIZACIÓN DE LA SENESCYT.....	159
FICHA DE LA SENESCYT	160

INDICE DE GRÁFICOS

ILUSTRACIÓN 1: SEXO DEL ENCUESTADO	38
ILUSTRACIÓN 2: EDAD DEL ENCUESTADO	38
ILUSTRACIÓN 3: INSTITUCIÓN ARMA SU HORARIO DE CLASES DEPENDIENDO DE LA DISPONIBILIDAD DEL DOCENTE.....	39
ILUSTRACIÓN 4: FORMA EN QUE EL DOCENTE CONOCE SU HORARIO	39
ILUSTRACIÓN 5: QUIÉN LE INFORMA EL HORARIO AL DOCENTE	40
ILUSTRACIÓN 6: MATERIAS QUE TIENE ASIGNADAS UN DOCENTE.....	40
ILUSTRACIÓN 7: DOCENTES DIRIGENTES DE CURSOS	41
ILUSTRACIÓN 8: IMPLEMENTACIÓN DE UN SISTEMA INFORMÁTICO QUE GENERA HORARIOS.....	41
ILUSTRACIÓN 9: DOCENTE LE GUSTARÍA TENER ACCESO AL SISTEMA WEB	42
ILUSTRACIÓN 10: TIEMPO DE ANTICIPACIÓN CON QUE EL DOCENTE CONOCE SU HORARIO.....	42
ILUSTRACIÓN 11: DÍAS DE ANTICIPACIÓN QUE EL DOCENTE SE INFORMA DE SU HORARIO	43
ILUSTRACIÓN 12: DISPOSICIÓN DEL DOCENTE PARA USAR EL SISTEMA WEB.....	43
ILUSTRACIÓN 13: CAUSA POR LA QUE NO USARÍA EL SISTEMA WEB	44
ILUSTRACIÓN 14: CAUSAS POR LAS QUE SI UTILIZARÍA EL SISTEMA WEB.....	45
ILUSTRACIÓN 15: RANKING DE LAS BASES DE DATOS MÁS USADAS EN HASTA JULIO 2016.....	52
ILUSTRACIÓN 16: ARQUITECTURA DE LA APLICACIÓN	53
ILUSTRACIÓN 17: CASOS DE USO DEL SISTEMA DE GENERACIÓN DE HORARIOS	57
ILUSTRACIÓN 18: CASOS DE USO DEL INGRESO AL SISTEMA.....	58
ILUSTRACIÓN 19: CASOS DE USO DE ADMINISTRACIÓN DEL SISTEMA	59
ILUSTRACIÓN 20: CASOS DE USO DE REGISTRO DE AÑO LECTIVO	60
ILUSTRACIÓN 21: CASOS DE USO DE REGISTRO DE MATERIAS	61
ILUSTRACIÓN 22: CASOS DE USO DE REGISTRO DE PROFESORES	62
ILUSTRACIÓN 23: CASOS DE USO DE ADMINISTRACIÓN DE CURSOS Y PARALELOS	63
ILUSTRACIÓN 24: CASOS DE USO DE REGISTRO DE PERFILES.....	64
ILUSTRACIÓN 25: CASOS DE USO DE REGISTRO DE USUARIOS.....	65
ILUSTRACIÓN 26: CASOS DE USO DEL PROCESO DE GENERACIÓN DE HORARIOS.....	66
ILUSTRACIÓN 27: CASOS DE USO DE CONSULTA DE REPORTES	67

INDICE DE TABLAS

TABLA 1: COMPARACIÓN DE LENGUAJES DE PROGRAMACIÓN	46
TABLA 2: CARACTERÍSTICAS DE LENGUAJES DE PROGRAMACIÓN	47
TABLA 3: COMPARACIÓN DE IDE'S	48
TABLA 4: CARACTERÍSTICAS DE FRAMEWORKS	49
TABLA 5: COMPARACIÓN DE BASES DE DATOS.....	50
TABLA 6: CARACTERÍSTICAS DE BASES DE DATOS	51
TABLA 7: DESCRIPCIÓN DEL CASOS DE USO DEL SISTEMA DE GENERACIÓN DE HORARIOS	57
TABLA 8: DESCRIPCIÓN DE LOS CASOS DE USO DEL INGRESO AL SISTEMA	58
TABLA 9: DESCRIPCIÓN DE CASOS DE USO DE ADMINISTRACIÓN DEL SISTEMA	59
TABLA 10: DESCRIPCIÓN DE CASOS DE USO DE REGISTRO DE AÑO LECTIVO	60
TABLA 11: DESCRIPCIÓN DE CASOS DE USO DE REGISTRO DE MATERIAS.....	61
TABLA 12: DESCRIPCIÓN DE CASOS DE USO DE REGISTRO DE PROFESORES.....	62
TABLA 13: DESCRIPCIÓN DE CASOS DE USO DE ADMINISTRACIÓN DE CURSOS Y PARALELOS	63
TABLA 14: DESCRIPCIÓN DE CASOS DE USO DE REGISTRO DE PERFILES	64
TABLA 15: DESCRIPCIÓN DE CASOS DE USO DE REGISTRO DE USUARIOS.....	65
TABLA 16: DESCRIPCIÓN DEL CASO DE USO DEL PROCESO DE GENERACIÓN DE HORARIOS.....	66
TABLA 17: DESCRIPCIÓN DEL CASO DE USO DE CONSULTA DE REPORTES	67
TABLA 18: FACTIBILIDAD TÉCNICA DE LAS HERRAMIENTAS	68
TABLA 19: ESPECIFICACIONES TÉCNICAS RECOMENDADAS DEL HOST	68
TABLA 20: ESPECIFICACIONES TÉCNICAS RECOMENDADAS DEL SERVIDOR DE SERVICIOS WEB	68
TABLA 21: FACTIBILIDAD ECONÓMICA DE LAS HERRAMIENTAS.....	69

RESUMEN

La solución propuesta en el presente proyecto de titulación, es el desarrollo e implementación del sistema web de generación automática de horarios académicos para la Unidad Educativa Federico González Suárez del cantón Duran, que nace a partir de la problemática del desgaste de tiempo de los recursos asignados durante la elaboración manual de los horarios de clases del año lectivo por aperturar, siendo uno de los objetivos del sistema optimizar el tiempo de elaboración de los horarios de clase a través de la automatización del mismo, considerando los cursos y paralelos aperturados, materias vigentes, tipo de aula que requiere la materia, materias por curso, materias que puede impartir el profesor, la disponibilidad del profesor y los periodos de clases de acuerdo a la duración de clases establecida sin considerar los recesos establecidos por el plantel, teniendo acceso a la visualización de la información del sistema de acuerdo al perfil del usuario correspondiente y desde cualquier dispositivo conectado a internet.

Palabras clave: HORARIOS; COLEGIO; ALGORITMO; DESARROLLO; SISTEMA WEB; CERTIFICADOS DIGITALES; FRAMEWORK; SERVIDOR WEB

ABSTRACT

The proposed solution in this undergraduate thesis project, is the development and implementation of high school timetable's automatic generation web system for Unidad Educativa Federico Gonzalez Suarez from Duran, which arises from the problematic of consuming human resources work time the during high school timetable's manual generation for the scholar year to start, being one of the objectives optimize the response time timetabling through the automation of process itself, considering grades levels, grades curriculum, subjects that a teacher can teach, teacher availability, class periods according to the class duration without considering breaks established by the education unit, having access to the required system information display, according to the profile associated to the corresponding user, from any device connected to internet.

Keywords: TIMETABLES; HIGH-SCHOOL; ALGORITHM; DEVELOPMENT; WEB SYSTEM; DIGITAL CERITFICATES; FRAMEWORK; WEB SERVER

INTRODUCCIÓN

Hoy en día, uno de los procesos que realizan las unidades educativas particulares, antes de empezar el año lectivo, es la elaboración de los horarios de clases. La elaboración de los horarios es un proceso manual en el que se conllevan varias horas para su realización, la cual se empieza a llevar a cabo una vez que se hayan considerado factores tales como la disponibilidad de tiempo de los profesores, la asignación materias por especialidad a los profesores, conocer cuál es la cantidad oficial aproximada de alumnos inscritos en el año lectivo para la elaboración, en caso de ser necesario, de paralelos en cada curso y de acuerdo a eso, la elaboración de horarios para cada curso.

Dentro del proceso de la elaboración de horarios, existen varios inconvenientes que surgen al momento de realizarlo de forma manual, uno de esos principales inconvenientes es el cruce de las horas clase de una materia entre dos o más cursos que la tengan asignada.

Por tal motivo en el presente trabajo de titulación se va a desarrollar un sistema web de generación automática de horarios para la unidad educativa Federico González Suárez, del Cantón Durán. La necesidad del presente sistema a desarrollar nace a partir del desgaste de tiempo durante la elaboración del horario de clases del año lectivo debido a que este se realiza de forma manual, por ende, una de las finalidades del sistema es optimizar el tiempo de elaboración del horario de clase a través de la automatización del mismo.

CAPITULO I: FUNDAMENTACIÓN CONCEPTUAL

1.1 Planteamiento del problema

La gestión y asignación de horarios académicos realizada por las autoridades respectivas de la Unidad Educativa Federico González Suarez del cantón Durán, es actualmente realizada manualmente siendo un proceso exhaustivo y analítico, el cual toma bastante tiempo de los recursos asignados, basándose en una revisión extenuante y tomando en consideración información necesaria como las materias organizadas por grados, las horas a la semana que se puede dar máximo por cada materia, el horario de disponibilidad del profesor, las materias que puede impartir cada docente, cantidad de aulas habilitadas según recursos necesarios y cantidad de asientos disponibles por aula. El proceso termina con la elaboración de hojas de cálculos de Excel donde se establecen los horarios de clases por cursos y paralelos, debido a que no existe un sistema automatizado para la gestión de horarios académicos en la Unidad Educativa mencionada.

1.2 Objetivo General

Desarrollar e implementar un sistema web gestor de horarios académicos automático para la Unidad Educativa Federico González Suarez del cantón Durán.

1.3 Objetivos Específicos

- Conocer la repercusión de elaboración manual de horarios académicos de la Unidad Educativa Federico González
- Realizar el levantamiento de información para conocer los requerimientos y delimitar la información académica del colegio a partir del proceso actualmente realizado.
- Analizar y diseñar el modelo entidad relación del sistema web de gestión de horarios
- Desarrollar e implementar el sistema web gestor de horarios académicos automáticos para la Unidad Educativa Federico González Suarez.

1.4 Alcance

El sistema web gestor de horarios académicos automáticos para la Unidad Educativa Federico González Suarez del cantón Duran, tendrá los siguientes módulos, y el acceso a cada uno de estos dependerá del perfil que tenga el usuario:

- Seguridad
 - Cambios de contraseña
- Mantenimiento
 - Año Lectivo
 - Materias
 - Profesores
 - Horario de profesores
 - Materias por profesores
 - Aulas
 - Usuarios
 - Perfil de usuarios
 - Permisos de perfiles
- Procesos
 - Mantenimiento de horarios académicos
- Reportes
 - Reporte de horarios académicos
 - Reporte de carga horaria de profesores
 - Reporte de profesores
 - Reporte de materias
 - Reporte de aulas
 - Reportes de asignación de materias
 - Reportes de horarios de los profesores

Los usuarios del sistema podrán acceder a los módulos y opciones correspondientes de acuerdo a los permisos que tenga, para lo cual se crearán tres perfiles de usuario:

- Administrador
- Coordinador
- Secretaria
- Profesor

El perfil Administrador tendrá acceso a cada una de las opciones indicada en los módulos sin excepción.

El perfil Coordinador tendrá acceso a todas las opciones del sistema excepto a la administración de los usuarios del sistema.

El perfil Secretaria tendrá acceso a todas las opciones del sistema excepto a la generación de los horarios académicos, ni a la administración de los usuarios del sistema.

El perfil Profesores tendrá acceso a ciertas opciones del módulo de reportes tales como Reporte de horarios académicos, Reporte de carga horaria de profesores, Reportes de asignación de materias.

1.4 Marco Teórico

Web

Se entiende por web al fenómeno social que surge a partir de las diversas aplicaciones en internet y los usos que a estas se les dan. De acuerdo a J. García y J.N Sánchez (2013):

Se trata de una etiqueta o concepto genérico que define una tendencia fundamentada en un impulso global por resaltar lo social en especial la participación y la colaboración en los diferentes ambientes: personal, social, laboral y educativo como opuesta a la tendencia anterior en la que el usuario era un sujeto pasivo. De modo que los usuarios 2.0 seleccionan, publican y comparten la información disponible a través de la interacción y cooperación, convirtiéndose de este modo, en los protagonistas del nuevo escenario. (p. 690)

Una de las características principales de la web es la comunicación entre individuos y dispositivos que se logra a través de esta. Esta característica forma parte de los cambios que nos indican Ruiz y Huertas (2014):

La web 2.0 es un cambio tecnológico que conlleva un nuevo uso de las tecnologías existentes en Internet. (...) Se caracteriza por la habilidad de las personas para colaborar e intercambiar información en línea, representando un gran cambio en la manera en que los individuos se comunican y colaboran con otros. (p. 65)

Es decir que se pueda acceder desde cualquier computador con conexión a internet mediante una dirección web o url que hace referencia al sitio que se quiera acceder y este a su vez deberá estar alojado en un determinado sitio donde se guardarán y actualizarán los respectivos archivos que dependiendo de su configuración serán de acceso interno o externo desde cualquier parte del mundo. Según Ruiz y Huertas (2014):

La web ha pasado de ser unidireccional, pasiva o de solo lectura, utilizándose principalmente como un medio para la difusión de información en un solo sentido con muy poca interacción con el usuario, a un modelo multidireccional o participativo, donde no sólo se conversa, sino que también se comparten contenidos en múltiples formatos (vídeos, textos, audios...), los cuales suelen ser generados por los propios usuarios. (p. 65)

No obstante, como parte de la comunicación que se lleva a cabo, se encierra una gran cantidad de datos y tipo de datos los cuales son la principal transmisión dentro de la web. De acuerdo a Barroso y Cabero (2013):

La web 2.0, que vendría caracterizada por diferentes aspectos, como por ejemplo: es dinámica ya que los contenidos se actualizan constantemente, es colaborativa pues se va elaborando con la participación de un colectivo de personas, las herramientas que se utilizan en la misma suelen ser simples e intuitivas, los programas pueden ser utilizados sin la necesidad de instalar ningún tipo de programa en el ordenador, se trabaja en la nube, ofrecen un

entorno amigable e interactivo, y asumir que lo importante no es la tecnología sino la persona que tiene la capacidad de gestionar qué, cuándo y cómo publicar y participar. (p. 77)

En base a dichos conceptos el término web encierra un conjunto de características enfocadas a las aplicaciones alojadas en internet, cuyo enfoque principal es la comunicación. Toda la información que se maneja o se consulta para la web posee una organización al momento de almacenarse o consultarse, es decir que se encuentra alojada en una dirección web.

Aplicación web

De acuerdo a Pilacuán (2014):

Una aplicación web es un conjunto de páginas que interactúan unas con otras y con diversos recursos en un servidor web, incluidas bases de datos. Esta interacción permite implementar características en su sitio como catálogos de productos virtuales y administradores de noticias y contenidos. Adicionalmente podrá realizar consultas a bases de datos, registrar e ingresar información, solicitudes, pedidos y múltiples tipos de información en línea en tiempo real. (p. 15)

Se entiende por aplicación web a un entorno de trabajo digital cuya finalidad es agilizar los procesos manuales efectuados en una empresa u organización, agrupando en diferentes módulos todas las actividades involucradas, categorizadas por subprocesos para la facilidad de interacción y ubicación de las opciones de la aplicación, permitiendo al usuario la posibilidad de gestionar y mantener toda la información organizada en un solo ambiente seguro y confiable que estará disponible las 24 horas del día para su utilización. Una de las ventajas de un aplicativo web es la accesibilidad desde cualquier computador con conexión a internet que los usuarios podrán interactuar desde los clientes (navegadores) ingresando a una dirección web que hace referencia al servidor web donde está

alojado el sitio, en el cual se guardarán y actualizarán los la información y archivos respectivos que dependiendo de su configuración serán de acceso interno o externo desde cualquier parte del mundo.

Framework

Framework o también denominado marco de trabajo es aquel paquete que se conforma de varios componentes los cuales estructuran la creación de aplicaciones. De acuerdo a Martínez (2014) “El término inglés framework concreta, en términos generales, un conjunto estandarizado de conceptos, prácticas y criterios para enfocar un problema en particular, que es de utilidad como referencia para afrontar y resolver problemas nuevos de características similares.”

Es decir que es un ambiente universal y reutilizable que proporciona todas las facilidades y requerimientos necesarios en un solo paquete, mismos que pueden ser adicionados o alterados a criterio del programador, para el desarrollo de un tipo de aplicación específico como por ejemplo una aplicación o sistema web.

Las estructuras de las aplicaciones que realizan tareas específicas, se basa en librerías, programas y plantillas ya existentes, es por eso que Moyota (2015) menciona que:

El término framework (marco de trabajo) se emplea en muchos ámbitos de desarrollo de aplicaciones no solo del tipo web, es así que pueden existir Frameworks para un sin número de ámbitos como, por ejemplo: aplicaciones médicas, desarrollo de juegos, virtualización de escenarios, interfaces de usuario, etc. (p. 20)

En la actualidad existen diversos frameworks que varían en diseño, funcionalidades implementadas reutilizables y patrón de arquitectura de software que hace referencia a la forma que se organizará, utilizará y desarrollará la aplicación. De acuerdo a Funes y Dasso (2014):

Un Framework para Aplicaciones Web es un tipo de framework diseñado para soportar el desarrollo de sitios dinámicos, aplicaciones, servicios y recursos para la web. Este tipo de frameworks provee por lo general toda la funcionalidad central a la mayoría de las aplicaciones web, tales como administración de la sesión del usuario, persistencia de datos, y sistemas de plantillas (templates) así como un conjunto de buenas prácticas y guías para el desarrollo.

Es por ello que se entiende como framework como aquella aplicación incompleta configurable que puede constituir una nueva aplicación que realice una nueva tarea en específico. Para la construcción de esa nueva aplicación se toma en cuenta también el hecho de que los componentes de los cuales va a estar integrada dicha aplicación depende el lenguaje de programación y el paradigma que se esté usando. Framework en programación es una abstracción en el cual el código común provisto, provee funcionalidades genéricas que pueden ser selectivamente anuladas o mejoradas por el código del usuario dándole funciones específicas para el desarrollo, son un caso especial de librerías de software porque que son abstracciones reutilizables de códigos envueltos en una muy bien definida interfaz de programación de aplicación, que contienen algunas características claves distintivas que los separan de las librerías normales

Sistema Gestor de Base de datos

Es un conjunto de datos que comparten una misma temática y son organizados, categorizados y almacenados digitalmente para su mejor administración evitando inconsistencia en la data ingresada y posterior reutilización considerando todas las medidas de seguridad necesaria para su acceso. De acuerdo a Menéndez (2015):

Un sistema gestor de almacenamiento de datos (o SGBD) es un conjunto de programas que permiten el almacenamiento, modificación y extracción de la información de la base de datos, además de proporcionar una serie de

herramientas que nos permiten realizar múltiples tareas como pueden ser añadir, borrar, modificar y analizar los datos. (p. 37)

Bajo la necesidad de gestionar, administrar y manejar los datos de la base de datos, Méndez (2015) explica que:

Un sistema gestor de bases de datos (SGBD) es una colección de datos interrelacionados y un conjunto de programas para acceder a dichos datos. La colección de datos conocida como base de datos, contiene información relevante para una empresa. El objetivo principal de un SGBD es proporcionar una forma de almacenar y recuperar la información de una base de datos de manera que sea tanto práctica como eficiente. Las bases de datos se diseñan para gestionar grandes cantidades de información. La gestión de los datos implica tanto la definición de estructuras para almacenar la información como la provisión de mecanismos para la manipulación de la información. Además, los sistemas de bases de datos deben garantizar la fiabilidad de la información almacenada. (p. 28)

Se concluye con la finalidad de una base de datos que, de acuerdo a Roche, Sarría y Cutiño (2014) define como:

EL propósito general de un Sistema de Base de Datos (SBD) es registrar e integrar la información, mantener su seguridad y brindar la posibilidad de acceder a la misma cuando se requiera. Para ello existen programas potentes denominados Sistemas de Gestión de Bases de Datos (SGBD) que pueden almacenar, organizar y manipular gran cantidad de información en una o varias bases de datos relacionales, manejar todas las solicitudes de accesos formuladas por los usuarios y facilitar el diseño e instalación. (p. 2)

Por ende, también se define como base de datos a la entidad donde se almacenan datos de una manera estructurada y a los cuales se tiene acceso. Las bases de datos son diseñadas para gestionar grandes cantidades de información y así mismo para la manipulación de los datos almacenados. Como característica

principal de una base de datos estructurada es que debe de existir la menor cantidad de redundancia en los datos.

Modelo Vista Controlador

Para la creación de aplicaciones, se debe de seguir un orden y una organización que es lo que propone el Modelo Vista Controlador (MVC), de acuerdo a Eslava (2013):

El Modelo Vista Controlador (MVC) es un patrón de arquitectura de software que separa los datos y la lógica de negocio de una aplicación de la interfaz de usuario y el módulo encargado de gestionar los eventos y las comunicaciones. Para ello MVC propone la construcción de tres componentes distintos que son el modelo, la vista y el controlador, es decir, por un lado, define componentes para la representación de la información, y por otro lado para la interacción del usuario. Este patrón de diseño se basa en las ideas de reutilización de código y la separación de conceptos, características que buscan facilitar la tarea de desarrollo de aplicaciones y su posterior mantenimiento. (p. 109)

MVC propone una forma de desarrollar un software de manera que sea más fácil de manejar para el usuario y pueda haber tanto reutilización como optimización de código, de acuerdo a Talledo (2015):

El modelo vista controlador (MVC) es un patrón de arquitectura de software que separa los datos y la aplicación de la interfaz de usuario y el módulo encargado de gestionar los eventos y las comunicaciones.

Para ello, MVC propone la construcción de tres componentes distintos que son el modelo, la vista y el controlador. El patrón de llamada y retorno MVC se ve, con frecuencia, en aplicaciones web. En este tipo de aplicaciones, la vista es la página HTML y el código que provee de datos dinámicos a la página. El modelo es el Sistema de Gestión de Base de Datos y la Lógica de negocio, y

el controlador es el responsable de recibir los eventos de entrada desde la vista. (p. 95)

Como se menciona en el párrafo que antecede, la finalidad principal del MVC es la organización y la estructura de una aplicación, y de acuerdo a Díaz y Fernández (2012):

El patrón Modelo-Vista-Controlador (MVC) surge con el objetivo de reducir el esfuerzo de programación, necesario en la implementación de sistemas múltiples y sincronizados de los mismos datos, a partir de estandarizar el diseño de las aplicaciones. El patrón MVC es un paradigma que divide las partes que conforman una aplicación en el Modelo, las Vistas y los Controladores, permitiendo la implementación por separado de cada elemento, garantizando así la actualización y mantenimiento del software de forma sencilla y en un reducido espacio de tiempo. A partir del uso de frameworks basados en el patrón MVC se puede lograr una mejor organización del trabajo y mayor especialización de los desarrolladores y diseñadores. (p. 47)

Por ello, se entiende al modelo vista controlador como la parte de arquitectura de software donde se separan los datos, la interfaz de usuario y la lógica de negocio, modelo el cual se aplica durante el desarrollo de un software, no obstante, se debe tener en cuenta la transición de los datos sin afectar al MVC logrando de esta manera que se cumpla el flujo y objetivo de la aplicación a desarrollar.

Algoritmo

Según Zaragoza y Trueba (2015) un algoritmo “Consiste en una técnica originada de la problemática de encontrar una alineación óptima entre dos secuencias de tiempo dadas bajo ciertas restricciones. Dichas secuencias son alineadas de manera no lineal con la finalidad de coincidir entre sí.” (p. 29), y de acuerdo a Guerrero y Barba (2008) “El algoritmo de selección propuesto está

basado en cinco componentes: la actividad del usuario, los requerimientos del servicio, las preferencias del usuario, las condiciones de la celda de acceso, y la función costo". (p. 195)

Partiendo del algoritmo, se enfoca la parte estructural de la aplicación en un modelo matemático, lo cual, de acuerdo a Álvarez y Ávila (2004):

Un modelo matemático no es más que un conjunto de relaciones funcionales que describen el comportamiento de un sistema, proceso o fenómeno según sea el caso. El mismo posee un número de variables que se vinculan entre sí por las relaciones funcionales. Entre las variables se encuentran algunas que por su vínculo dentro del sistema ya tienen un valor fijo, por tal razón se les conocen como variables fijas. Las que no tienen un valor fijo pueden ser consideradas como variables libres y como variables de estado. Las variables libres toman su valor libremente, por conveniencia. Las variables de estado se determinan por despeje o cálculo en una relación funcional del modelo, en función de las variables fijas y libres.

En base a dichos conceptos, se puede definir como algoritmo al conjunto ordenado de procedimientos y reglas que permiten realizar una operación o cálculo para llegar a la solución de un tipo de problemas, recibiendo datos de varios tipos como parámetros de entrada para poder realizar el análisis y la solución del problema.

Sistema Gestor

Como complemento adicional a la organización de la aplicación se aplica el rol de gestor, cuyo cargo según Aguilar, Rivas y Cerrada (2014):

Este agente se encarga de ubicar las aplicaciones que puedan ser requeridas por un proceso que se esté ejecutando, como por ejemplo de acceso a redes, programas de cálculo numérico o simbólico, aplicaciones de inteligencia artificial, de envío y recepción de mensajes, etc. Dichas aplicaciones pueden

estar en cualquier servidor al que se tenga acceso y son requeridas por los agentes de bases de datos, de administración de recursos, y algunos de los agentes de la comunidad de agentes de control del SCDIA (coordinadores y especializados).

Como partes de las funcionalidades de dicho rol, Arnau y Martín (2013) indica que “Este rol permite acceder a todas las funcionalidades del sistema de gestión y realizar todas las configuraciones posibles, entre ellas, la asignación de roles a los diferentes usuarios del sistema y la introducción de espacios y materiales” (p. 59). En base a dichos conceptos, se define como gestor como la parte del sistema operativo que permite visualizar aquellas aplicaciones que se encuentran ejecutando tanto en primer como en segundo plano y los recursos que estas utilizan. Parte también de sus funcionalidades el permitir el almacenamiento, modificación y eliminación de los datos de una base de datos, además de dar las herramientas necesarias para poder manipular los mismos.

Software

La aplicación a crear debe de abarcar el concepto de software, cuyo concepto de acuerdo a Flores (2013) se define como “Un conjunto de programas destinados a permitir la comunicación del usuario con un ordenador y gestionar sus recursos de manera eficiente” (2013) Dicho concepto concuerda con Solinas y Antonelli (2013):

El término “software” tal cual hoy lo conocemos, fue utilizado por primera vez por John W. Tukey en 1957. Tukey fue un estadístico que acuñó muchos términos que ahora son de uso común, pero las dos palabras más famosas inventadas por él están relacionadas con la informática. Mientras trabajaba con John von Neumann en los primeros diseños de computadoras, Tukey introdujo la palabra "bit" como contracción de "Dígito binario" (Binary Digit). Tukey usó el término “Computer Software” en un contexto computacional en un artículo de 1958 en “American Mathematical Monthly”.

Como dato adicional existe las expresiones de software conocidas como software propietario, cuyo uso, según Rodríguez (2012) es:

La expresión “software propietario” proviene del término en inglés “proprietary software”. En la lengua anglosajona, “proprietary” significa poseído o controlado privadamente (privately owned and controlled), que destaca la mantención de la reserva de derechos sobre el uso, modificación o redistribución del software. Se debe aclarar que el término “propietario” es el inicialmente utilizado, pero con el inconveniente que la acepción proviene de una traducción literal del inglés, no correspondiendo su uso como adjetivo en el español, de manera que puede ser considerado como un barbarismo. Sin embargo, todavía es el término preferido por cerca del 73% de los sitios en Internet. Razón por la cual nos referiremos al software como propietario, pero también es válida la denominación de “dominio privado”, indicando que esta última acepción es más apropiada.

Por ello se concluye como el término software como la aplicación desarrollada en un lenguaje de programación para realizar una actividad específica, lo cual se va a implementar en dicho trabajo de investigación haciendo uso de un algoritmo matemático para la generación automática de horarios académicos.

Certificado digital SSL

En lo que concierne a certificado digital SSL, de acuerdo a Zubieta (2015) “SSL (Secure Sockets Layer) es un protocolo que se utiliza para el intercambio seguro de información. Así, junto http con SSL resulta https, un protocolo para poder navegar de manera segura por internet.”

Una navegación segura garantiza que toda la información sensible que viaje a través de internet, aunque pueda interceptada, no logre ser interpretada de forma alguna por algún intruso, lo que genera seguridad y confiabilidad en el proceso y al cliente con respecto a la información proporcionada.

SSL maneja dos claves para su proceso de encriptación, una pública de lado del cliente y otra privada que conocen tanto el emisor como el receptor para poder descifrar los mensajes correspondientes, tal como lo menciona Vásquez (2014):

Secure Socket Layer, el protocolo de capa de conexión segura a SSL y Transport Layer Security seguridad de la capa de transporte o TLS, su sucesor, son protocolos criptográficos que proporcionan comunicaciones seguras por una red, comúnmente Internet, SSL proporciona autenticación y privacidad de la información entre extremos sobre internet mediante el uso de criptografía, habitualmente solo el servidor es autenticado, se garantiza su integridad mientras el cliente se mantiene sin autenticar, la autenticación mutua requiere un despliegue de infraestructura de claves públicas para los clientes, los protocolos permiten a las aplicaciones cliente-servidor comunicarse de una forma diseñada para prevenir las escuchas, la falsificación de la identidad del remitente y alterar la integridad del mensaje. (p. 20)

Lo que se entiende por certificado digital es aquel documento electrónico que confirma el uso de una clave para un usuario, tal como lo definen Gaona y Montenegro (2014):

Un certificado digital es un documento electrónico por medio del cual se puede dar fe de que una determinada clave pública pertenece a un determinado usuario. La institución que garantiza esta relación entre usuario y clave pública es denominada autoridad de confianza (CA). Para garantizar que el certificado digital no ha sido alterado, este documento va firmado con la clave privada de una CA, de manera tal que cualquier usuario del sistema puede verificar la validez del certificado con la clave pública de la CA. La misión de las CA es generar el par de claves de los usuarios y mantener un registro de claves públicas, para que cualquier usuario pueda verificar la firma de otro usuario con ayuda de la CA. (p. 55)

Concordando la definición de certificado digital con Gómez, Candela y Sepúlveda (2013) “un certificado digital es un documento digital mediante el cual un tercero

confiable o autoridad certificadora garantiza la vinculación entre la identidad de un sujeto o entidad y su llave pública” (pp. 34-35). De acuerdo a lo antes mencionado, se entiende por certificado digital al documento electrónico que, valida el uso de claves para la transferencia de datos, y dicha clave es de conocimiento del emisor y receptor de la información.

Algoritmo de cifrado

Existen varios algoritmos de cifrados, uno de ellos es el algoritmo AES que, de acuerdo a Pousa, Sanz y De Giusti (2015):

AES (Advanced Encryption Standard) se caracteriza por ser un algoritmo de cifrado por bloques. Los datos a encriptar se dividen en bloques de tamaño fijo (128 bits), donde cada bloque se representa como una matriz de 4x4 bytes llamada estado. A cada estado se le aplican once rondas, cada una está compuesta por un conjunto de operaciones. Las once rondas se pueden clasificar en tres tipos: una ronda inicial, nueve rondas estándar y una ronda final. (p. 3)

Para complementar las medidas de seguridad en el cifrado de la contraseña se utilizará también la función hash, que de acuerdo a Blain y Vazquez (2012):

Una función hash es una función que transforma una cadena binaria o mensaje de longitud arbitraria (por ejemplo, x) en otra cadena binaria (por ejemplo, Z) de longitud constante, generalmente más pequeña, que es la salida de la función, llamada un valor hash y son empleados para la emisión de certificados, firmas digitales, generación de claves y verificación de password. Este resumen o función hash debe ofrecer ciertos niveles de seguridad para garantizar la integridad del mensaje y pueda ser empleado en criptografía (p. 1)

Esta función emplea algoritmos específicos entre los cuales Blain y Vazquez (2012) mencionan:

MD5: Desarrollado por Ron Rivest en 1992. Es una mejora al MD4 y MD2 (1990), es más lento, pero con mayor nivel de seguridad. Resumen de 128 bits. Aunque está obsoleto desde 2005 su algoritmo constituye la base de otras funciones y SHA-1: Desarrollado por el NIST, National Institute of Standards and Technology, 1995. Similar a MD5 pero con resumen de 160 bits. Existen otras propuestas conocidas como SHA-256 y SHA-512, posibles estándares. (p. 2)

Se entiende por algoritmo de cifrado al algoritmo implementado en la transmisión de información haciendo uso de un conjunto de claves para cifrar la información transmitida, esto con la finalidad de evitar la interceptación de dicha información y cuyo descifrado solo pueda ser aplicado por el receptor. Con la finalidad de asegurar la privacidad de las contraseñas se utilizará una mezcla de los algoritmos AES Y SHA1 lo cual es considerado criptográficamente más seguro y óptimo para el siguiente proyecto.

Servidor web

Una definición sencilla de servidor web es la que plantea Ramos (2014) “Un servidor web es un programa diseñado para aceptar peticiones HTTP del navegador y servir las páginas web que tiene alojadas.” (p. 28) En concordancia a lo indicado además del objetivo de un servidor web, Poveda, Herrera y Torres (2016) indican lo siguiente:

Para poder visualizar la aplicación web en el internet, es necesario alojar los archivos de la misma en un servidor web, mayormente conocido como web server. Un web server es tanto el ordenador como el programa que implementa el protocolo HTTP (Hypertext Transfer Protocol - Protocolo de transferencia de hipertexto), diseñado para transferir lo que llamamos hipertextos, páginas Web o páginas HTML. El navegador de un cliente particular realiza una petición al servidor y éste le responde. (p. 4)

Pilacuán (2014) indica que:

Un servidor web puede ser tanto un ordenador de grandes dimensiones y capacidad como un programa informático que utiliza el protocolo de comunicaciones http para recibir las peticiones de información de un programa cliente (navegador) en el ordenador del usuario. Una vez interpretada la solicitud que el cliente hace, el servidor web recupera la información solicitada de sus carpetas y la envía al programa cliente para su visualización. Cuando los servidores web utilizan protocolos de seguridad para la transmisión de datos entre ellos y los clientes, hablamos de servidores web seguros. (p. 19)

Se entiende entonces por servidor web al servidor en el cual se encuentra alojada toda la información de una página o aplicación web y la cual es necesaria para su correcto funcionamiento, como característica principal se conoce que utiliza el protocolo HTTP.

CAPITULO II: METODOLOGIA DE LA INVESTIGACION

2.1 Tipo de Investigación

El tipo de investigación aplicada en el presente trabajo de titulación es la investigación descriptiva. De acuerdo a Ruiz (2011) “la investigación descriptiva tiene como objetivo primordial la descripción de la realidad, siendo sus principales métodos de recogida de información la encuesta e incluso la observación” (p. 191).

El motivo por el cual es aplicada dicho tipo de investigación se debe a que se realiza un análisis y recolección de datos, tales como la disponibilidad de los profesores, para conocer como intervienen en la generación del horario de clases así como también cuáles son las condiciones para que este se pueda construir.

2.2 Enfoque Metodológico

El enfoque del presente trabajo de titulación es la metodología cualitativa, esta investigación de acuerdo a Krause se define como los “procedimientos que posibilitan una construcción de conocimiento que ocurre sobre la base de conceptos. Son los conceptos los que permiten la reducción de complejidad y es mediante el establecimiento de relaciones entre estos conceptos que se genera la coherencia interna del producto científico” (1995, pág. 3).

Los componentes que conforman la investigación cualitativa son los datos y los procedimientos de acuerdo a lo indicado por Strauss, Corbin y Zimmerman (2002), quien define a los datos como los que “pueden provenir de fuentes diferentes, tales como entrevistas, observaciones, documentos, registros y películas” (p. 13). En lo que respecta a los procedimientos, Strauss, Corbin y Zimmerman lo definen como lo que “los investigadores pueden usar para interpretar y organizar los datos. Entre estos se encuentran: conceptualizar y reducir los datos, elaborar categorías en términos de sus propiedades y dimensiones, y relacionarlos, por medio de una serie de oraciones proposicionales” (p. 13).

En base a lo indicado anteriormente, se utiliza la investigación cualitativa para realizar el sistema web de generación automática de horarios de clase de la Unidad Educativa Federico González Suárez, debido a que se utilizará una entrevista para la obtención de información para el desarrollo del sistema.

El objetivo principal de la investigación cualitativa en el presente trabajo es dar a conocer el funcionamiento y la relación de los componentes del horario de clase, cuya finalidad es de que no existan inconsistencia ni redundancia en los datos finales que lo conformen.

La metodología que se utiliza en este proyecto de titulación desde el punto de vista de desarrollo de software es el modelo incremental, porque se aplica a las necesidades del proyecto, debido a que se han especificado desde el inicio, el modelo cascada representa una estructura secuencial de las etapas del software, es decir ordena las etapas del ciclo de vida del software de tal forma que cada fase del mismo debe terminar antes de poder continuar con la siguiente.

El ciclo de vida que vamos a utilizar para el desarrollo del sistema web que generara los horarios automáticos consiste primero en la fase de análisis en el cual se va a analizar todos los requerimientos para poder implementar el sistema web, la siguiente fase que se realizara es el diseño donde se definirá todos los diagramas y estructuras que se utilizará, luego se realizará el desarrollo en cual se codificará todos los requerimientos detallados en la fase del análisis y se concluirá con la fase de pruebas.

2.3 Instrumento

De acuerdo a Hernández, Fernández y Baptista (2006) la investigación cualitativa posee dispone de “diversos tipos de instrumentos para medir las variables de interés y en algunos casos llegan a combinarse varias técnicas de recolección de los datos” (p. 309). Los instrumentos que Hernández Sampieri, Fernández y Baptista mencionan son los cuestionarios, entrevistas de preguntas pre codificadas,

entrevistas de preguntas abiertas, entre otros, siendo esta, la última mencionada, la que se utiliza en el presente trabajo.

Como parte de los datos que son utilizados para la elaboración de horarios, se realiza el levantamiento de información por medio de una entrevista compuesta de 18 preguntas, obteniendo como variables principales las horas clase, las materias y los profesores.

Los datos adicionales que se obtienen a través de la entrevista son las distribuciones de horas clase por materia y la distribución de materias por profesor, adicionando el promedio de horas clases y de horas libres por profesor.

Otra técnica que se utiliza en este trabajo de titulación es la encuesta dirigida a los profesores de la Unidad Educativa para determinar la aceptación de la implementación del sistema web para la generación de horarios académicos, la encuesta está conformada por 14 preguntas cerradas.

2.4 Aplicación y Procesamiento: Herramientas estadísticas

Para la tabulación y procesamiento de los datos para este trabajo de titulación se utilizó la herramienta de Microsoft Excel a través de la cual se obtuvieron los gráficos estadísticos.

El universo para realizar las encuestas a los docentes de la Unidad Educativa Federico González Suarez es el 100%, debido a que el número de docentes de esta unidad educativa no es muy extenso, razón por la cual la muestra poblacional está conformada por 21 docentes.

2.5 Análisis de los resultados

Pregunta uno: Sexo del encuestado

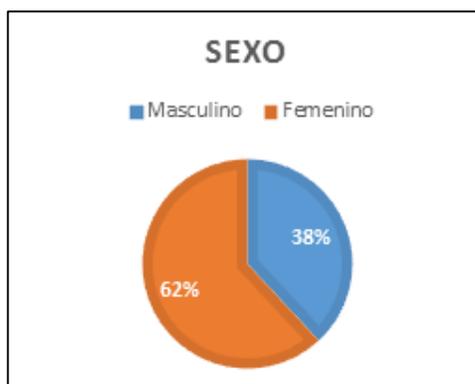


Ilustración 1: Sexo del encuestado

Fuente: Elaboración propia

El 62% de los docentes de la Unidad Educativa Federico González Suarez pertenecen al sexo femenino, y solo el 38% son del sexo masculino.

Pregunta dos: Edad del encuestado

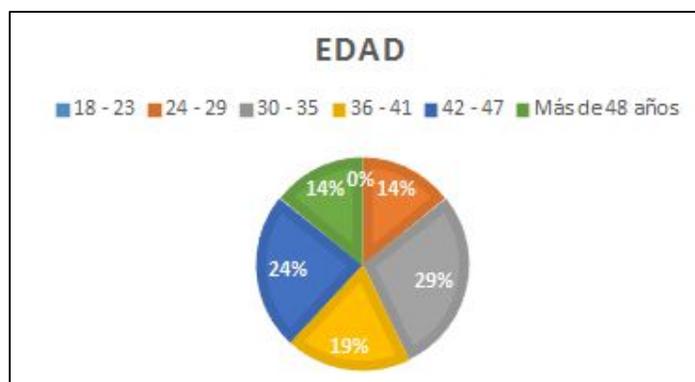


Ilustración 2: Edad del encuestado

Fuente: Elaboración propia

Como resultados de la encuesta se obtiene que el 29%, es decir la mayoría de los docentes encuestados su edad fluctúa entre un rango de 30 a 35 años.

Pregunta tres: ¿La institución arma su horario de clases conforme a su disponibilidad?



Ilustración 3: Institución arma su horario de clases dependiendo de la disponibilidad del docente

Fuente: Elaboración propia

El 71% de los docentes considera que la Unidad Educativa arma su horario de clases dependiendo de su disponibilidad, sin embargo, un 29% indica que su horario no se encuentra conforme a su disponibilidad.

Pregunta cuatro: ¿De qué manera le dan a conocer su horario de clases?

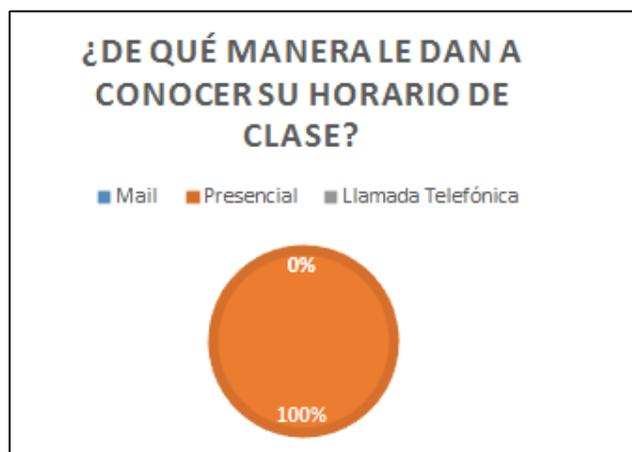


Ilustración 4: Forma en que el docente conoce su horario

Fuente: Elaboración propia

Como resultado de esta encuesta se puede observar que el 100% de los encuestados recibe su horario de clases de forma presencial.

Pregunta cinco: ¿Quién le da a conocer el horario académico?



Ilustración 5: Quién le informa el horario al docente
Fuente: Elaboración propia

Se puede observar que del 100% de los docentes encuestados, el 81% de ellos indican que el inspector es quién les informa su horario académico de forma presencial, al 10% le informa el vicerrector y solo al 9% le informa la secretaria.

Pregunta seis: ¿Cuántas materias tiene asignadas actualmente?



Ilustración 6: Materias que tiene asignadas un docente
Fuente: Elaboración propia

Como resultados de esta encuesta se obtiene, que del 100% de los docentes encuestados, la mayoría, 62%, tienen asignadas entre 1 y 3 materias, mientras que apenas el 9% tiene entre 7 y 9 materias.

Pregunta siete: ¿Es dirigente de algún curso?



Ilustración 7: Docentes dirigentes de cursos
Fuente: Elaboración propia

Del 100% de los encuestados sólo el 29% de ellos indican que no son dirigentes de curso.

Pregunta ocho: ¿Está de acuerdo con la implementación de un sistema informático que genera los horarios de clase de forma automática, considerando su horario de disponibilidad?

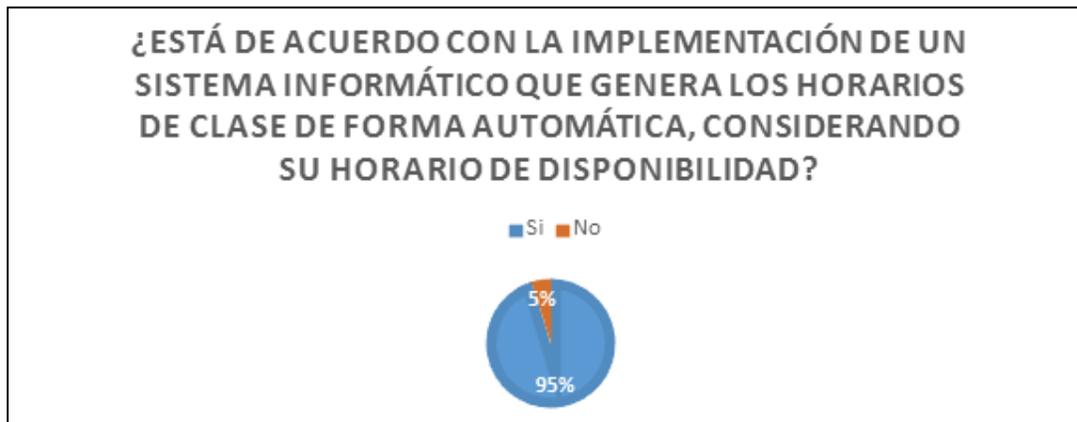


Ilustración 8: Implementación de un sistema informático que genera horarios
Fuente: Elaboración propia

Según esta encuesta el 95% de los encuestados estaría de acuerdo con la implementación del sistema web que genere los horarios académicos de la Unidad Educativa Federico González Suarez.

Pregunta nueve: ¿A través del internet, le gustaría tener acceso al sistema informático para conocer su horario de clase desde cualquier sitio?

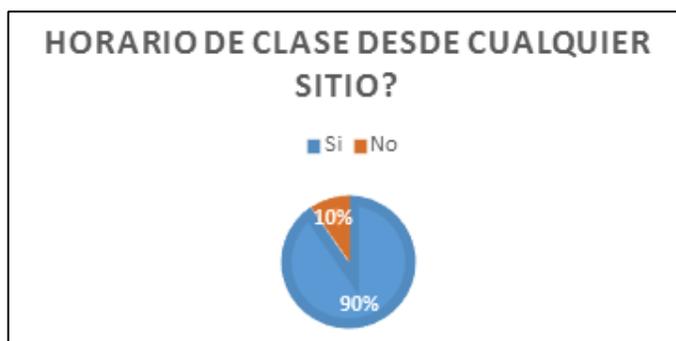


Ilustración 9: Docente le gustaría tener acceso al sistema web
Fuente: Elaboración propia

Al 90% de los docentes encuestados les gusta la idea de poder acceder al sistema web para conocer el horario académico asignado.

Pregunta diez: ¿Le informan con anticipación su horario de clase?



Ilustración 10: Tiempo de anticipación con que el docente conoce su horario
Fuente: Elaboración propia

Según los resultados obtenidos de esta encuesta el 81% de los docentes conoce sus horarios académicos con anticipación.

Pregunta once: ¿Con qué tiempo de anticipación le informan su horario de clases?



Ilustración 11: Días de anticipación que el docente se informa de su horario
Fuente: Elaboración propia

El 67% de los docentes de la Unidad Educativa se informa de los horarios en un rango de 1 a 7 días, el 24% de los encuestado se informa de su horario académico con más de 8 días de anticipación y sólo el 9% se informa el mismo día que empieza el año lectivo.

Pregunta doce: ¿Qué tan dispuesto estaría usted en utilizar el sistema informático para consultar sus horarios académicos por Internet?

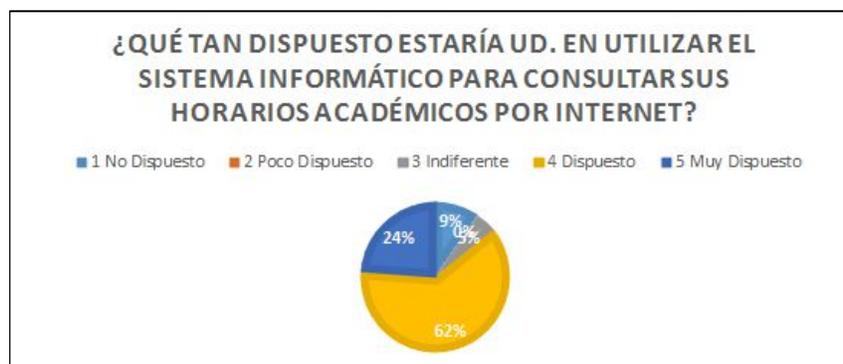


Ilustración 12: Disposición del docente para usar el sistema web
Fuente: Elaboración propia

El 62% y 24% de los encuestados estaría dispuestos y muy dispuestos respectivamente, es decir el 86% estaría dispuesto a acceder al sistema web para consultar los horarios académicos.

Pregunta trece: ¿Por qué no utilizaría el sistema informático para consultar los horarios académicos?



Ilustración 13: Causa por la que no usaría el sistema web
Fuente: Elaboración propia

El 43% de los encuestados no utilizaría el sistema web por el motivo de no tener acceso a internet, el 38% de los docentes señaló otras causas, el 10% no accedería por seguridad, y el 9% porque no tiene los medios para acceder al internet.

Pregunta catorce: ¿Por qué si utilizaría el sistema informático para consultar los horarios académicos?



Ilustración 14: Causas por las que si utilizaría el sistema web
Fuente: Elaboración propia

El 67% de los docentes encuestados sí utilizaría el sistema web para ahorrar tiempo.

CAPITULO III: PROPUESTA

3.1 Herramientas de desarrollo

De varias herramientas de desarrollo que existen en el mercado, se ha escogido a Java/Ajax como lenguaje de programación, la versión Luna de Eclipse for Java EE Developers como el IDE a usarse y MySQL como el motor de base de datos.

3.1.1 Lenguaje programación

Para realizar la selección del lenguaje de programación a usar se elaboró un cuadro comparativo de tres lenguajes de programación, que se detalla en la tabla 1, en la cual se pusieron las principales especificaciones técnicas de AJAX, ASP y PHP.

	JAVA/AJAX	ASP	PHP
Empresa	Oracle	Microsoft	The PHP Group
Costo	No	Si	No
Sintaxis de lenguaje base	JavaScript XML	VB C#	C/C++
Orientado a objetos	Parcialmente	Si	Si
Sistemas Operativos	Windows Linux	Windows Linux	Windows Linux
Servidor	Apache Tomcat Glassfish JBoss	IIS Mono	Apache
Base de datos	Oracle MySQL	SQL Server	MySQL SQL Server Oracle

Tabla 1: Comparación de lenguajes de programación

Fuente: Elaboración propia

Se consideran las especificaciones del cuadro anterior las necesarias para poder realizar la comparación entre los lenguajes de programación mencionados en

donde se aprecia la similitud en la compatibilidad de los sistemas operativos, pero se evidencian grandes diferencias entre las cuales resalta la sintaxis del lenguaje base de programación y la compatibilidad de servidores que posee JAVA/AJAX. En la tabla 2 se describen las características de los tres lenguajes de programación:

Características	
AJAX	<ul style="list-style-type: none"> • Las aplicaciones que se desarrollan usando AJAX permiten mayor interacción y navegación dentro de ella. • Permite elaborar un diseño similar a aplicaciones de escritorio • El flujo de información es inferior al de otros lenguajes • El procesamiento de datos se realiza en la parte del cliente liberando memoria del servidor. • Permite manejar la página por medio de layouts para poder realizar actualizaciones de sectores de la página y no requerir la actualización de la página completa
ASP	<ul style="list-style-type: none"> • ASP solamente es gratuito para Microsoft Windows NT o Windows 95/98. • El código ASP se puede mezclar con código HTML y puede ser editado en un bloc de notas. • El procesamiento de datos se realiza del lado del servidor incrementando el uso de memoria del servidor. • Posee mayor compatibilidad en el manejo de motores de bases de datos. • Da la facilidad de utilizar opciones de aplicaciones de escritorio en el desarrollo web
PHP	<ul style="list-style-type: none"> • Es de código abierto. • Es multiplataforma. • Soporte para varios servidores web. • Fácil acceso a distintos motores de bases de datos. • Posee integración con apache server. • PHP está orientado a objetos

Tabla 2: Características de lenguajes de programación

Fuente: Elaboración propia

El motivo por el cual se ha escogido AJAX como lenguaje de programación para este trabajo de titulación es porque utiliza estándares abiertos reconocidos que simplifican el tiempo de desarrollo del sistema utilizando lenguajes comunes y fáciles de implementar, es mucho más rápido en cuanto a carga de información, además de lograr la optimización de recursos debido a que no se recarga la página

frecuentemente, tiene la característica principal de poder actualizar partes de la página sin necesidad de realizar la actualización total de la página gracias a las peticiones asíncronas que no bloquean el cliente y le permite realizar otras operaciones.

3.1.2 IDE

Para realizar la selección del IDE a usar se elaboró un cuadro comparativo de tres IDE's, en el cual se pusieron las principales características de Eclipse Luna, Netbeans 8.1 e IntelliJ Community, el cual se describe en la tabla 3:

	Eclipse Luna	Netbeans	Intelij IDE
Versión	4.4	8.0.2	Community Edition
Lenguajes	Java ANSI C C++ JSP sh PHP PERL	Java PHP C/C++ Groovy	Java Groovy Kotlin XML JSON YAML
Frameworks	Oracle ADF EclipseLink Kodo Struts JSF ZK	Java FX Web and Java EE ZK	Android Swing JavaFX
Sistema Operativo	Windows Linux Mac	Windows Linux Mac	Windows Linux Mac
Portable	Si	No	No
Procesador de SO recomendado (Microsoft)	Intel Core i5	Intel Core i5	Intel Core i3
Memoria de SO recomendada (Microsoft)	2 GB	2 GB (32-bit) 4 GB (64-bit)	2 GB
Espacio en disco recomendado (Microsoft)	2 GB	1.5 GB	300 MB

Tabla 3: Comparación de IDE's

Fuente: Elaboración propia

El IDE escogido para la elaboración del sistema web es la versión Luna de Eclipse IDE for Java EE Developers, el cual es un IDE de programación el cual se compone de un conjunto de herramientas para desarrollar proyectos de código abierto. Su utilidad nace de sus siglas Integrated Drive Electronics.(IDE) que es en este caso utiliza el IDE de Java JDT (Java Development Toolkit) con su respectivo compilador.

3.1.3 Frameworks

Para realizar la selección del framework a usar se elaboró un cuadro comparativo de tres frameworks, el cual se detalla en la tabla 4, en el se pusieron las principales características de ZK, VAADIN y JSF.

Características	
ZK	<ul style="list-style-type: none"> • Da facilidad para el diseño de interfaz de usuario. • No es necesario el conocimiento previo de Ajax o JavaScript. • Está basado en componentes dirigidos por eventos. • Permite centrar toda la lógica de programación del lado del servidor. • Framework de código abierto • Interfaz más amigable
VAADIN	<ul style="list-style-type: none"> • Soporta la mayoría de navegadores web. • Está basado en GWT del lado del cliente. • Es compatible con los modelos MVC y MVP. • Posee una gran variedad de componentes para la interfaz de usuario.
JSF	<ul style="list-style-type: none"> • Forma parte del estándar J2EE. • Las etiquetas que se usan en el lenguaje son similares al código HTML. • Se pueden realizar validaciones dentro del código. • Se pueden realizar configuraciones en formato XML.

Tabla 4: Características de Frameworks

Fuente: Elaboración propia

El framework a utilizar es ZK el cual es un framework de código abierto que facilitará la creación y el diseño de la interfaz de usuario. Está basado en Java utilizando AJAX para el desarrollo de aplicaciones web similar al desarrollo de

aplicaciones de escritorio. Cuenta con una amplia gama de componentes para el desarrollo de la interfaz de usuario.

3.2 Base de Datos

Para realizar la selección del motor de base de datos a usar se elaboró un cuadro comparativo de tres motores de bases de datos, en la tabla 5 se describen las principales especificaciones de MySQL, Oracle y SQL Server:

	MySQL	Oracle	SQL Server
Empresa/ Corporation	Oracle	Oracle	Microsoft
Costo	NO	SI	SI
Licencia	GPL	Privada	Privada
Versión Express	-	Oracle Express	SQL Server Express (límite 10 GB)
Última Versión	MySQL Workbench 6.3.7	Oracle Database 12c Release 1	SQL Server 2016
Sistema Operativo	Linux Windows	Unix Windows	Windows
Memoria RAM recomendada	512 MB	1 GB	2 GB
Protocolo de red	TCP/IP	TCP/IP	TCP/IP

Tabla 5: Comparación de Bases de datos

Fuente: Elaboración propia

Se consideran las especificaciones del cuadro anterior necesarias para poder realizar la comparación entre los motores de bases de datos, se pueden observar varias similitudes entre ellos, como la compatibilidad con los sistemas operativos y el protocolo de red a usar, así como también se visualizan grandes factores diferenciadores como por ejemplo la licencia y sus rangos de costos. En la tabla 6 se describen las características principales de estos tres motores de base de datos:

Características	
MySQL	<ul style="list-style-type: none"> • Aprovecha la potencia de sistemas multiprocesador, gracias a su implementación multi-hilo. • Soporta una gran variedad de tipos de datos. • Soporta hasta 32 índices por tabla. • Permite la administración de credenciales. • Es un motor de base de datos de código abierto. • Es una de las herramientas más utilizadas por los programadores orientados a Internet. • Es el motor de base de datos considerado como el más rápido de Internet
Oracle	<ul style="list-style-type: none"> • Utilización de constantes, variables y tipos. • Permite el uso de procedimientos anónimos. • Posee funciones abstractas almacenadas. • Manejo de errores por medio de excepciones. • Se pueden insertar comentarios en el código
SQL Server	<ul style="list-style-type: none"> • Compatibilidad con la mayoría de las tareas administrativas de SQL Server. • Un entorno único integrado para la administración del motor de base de datos de SQL Server. • Usa cuadros de diálogos para el manejo de mensajes de alerta

Tabla 6: Características de Bases de datos

Fuente: Elaboración propia

El motor de base de datos a usar es MySQL el cual es un motor de base de datos de código de abierto, además de ser un motor de alto rendimiento y estar basado en aplicaciones web, se encuentra dentro de los tres primeros puestos del ranking de bases de datos relacionales de acuerdo a (DB-Engines, 2016) que lo ubica en segundo puesto de acuerdo al modelo de calificación y clasificación propuesto y descrito en la página.

Rank			DBMS	Database Model	Score		
Jul 2016	Jun 2016	Jul 2015			Jul 2016	Jun 2016	Jul 2015
1.	1.	1.	Oracle	Relational DBMS	1441.53	-7.72	-15.20
2.	2.	2.	MySQL +	Relational DBMS	1363.29	-6.85	+79.95
3.	3.	3.	Microsoft SQL Server	Relational DBMS	1192.89	+27.08	+89.83
4.	4.	4.	MongoDB +	Document store	315.00	+0.38	+27.61
5.	5.	5.	PostgreSQL	Relational DBMS	311.15	+4.55	+38.33
6.	6.	6.	DB2	Relational DBMS	185.08	-3.49	-13.04
7.	7.	↑ 8.	Cassandra +	Wide column store	130.70	-0.42	+17.99
8.	8.	↓ 7.	Microsoft Access	Relational DBMS	124.90	-1.32	-19.40
9.	9.	9.	SQLite	Relational DBMS	108.53	+1.75	+2.66
10.	10.	10.	Redis +	Key-value store	108.03	+3.54	+12.96

Ilustración 15: Ranking de las bases de datos más usadas en hasta julio 2016
Fuente: (DB-Engines, 2016)

3.3 Arquitectura de la solución

La arquitectura utilizada para el desarrollo del sistema planteado es de Cliente/Servidor de 3 capas, logrando de esta forma dividir la carga y procesamiento del sistema en tres partes de acuerdo a sus funciones:

Cliente, donde se realiza la presentación de la interfaz de usuario con el que se interactúa con el sistema, para aplicaciones web es mediante un navegador web que ayuda en la carga de una página específica.

Aplicación, donde se encuentra toda la lógica del negocio del sistema y procesamiento de la información, sirve de puente entre el cliente y el servidor pasando primero por una validación de datos ingresados por el cliente para posteriormente ser enviados al servidor.

Servidor, donde se encuentra toda la información del sistema y por ende la encargada del acceso a los mismos, en caso de recibir una petición de recuperación o consulta y por mantenimiento como ingreso, modificación y/o eliminación de contenido de la base de datos.

Dentro de las ventajas del uso de este tipo de arquitecturas se encuentra el concepto de centralización debido a que une diferentes conceptos y herramientas independientes entre sí en un solo sistema, lo cual ayuda al momento de desarrollo y mantenimiento del mismo debido a que puede ser programado en cualquier lenguaje, además disminuye las conexiones a base de datos debido a que el acceso a este no se realiza directamente desde la interfaz de usuario sino desde la capa

intermedia de aplicación, acogiendo todas las peticiones recibidas como un solo usuario, y como se separa en diferentes capas la información y lógica del sistema del cliente lo hace menos vulnerable a posibles ataques con buenas medidas de seguridad implementadas previos al acceso del servidor.

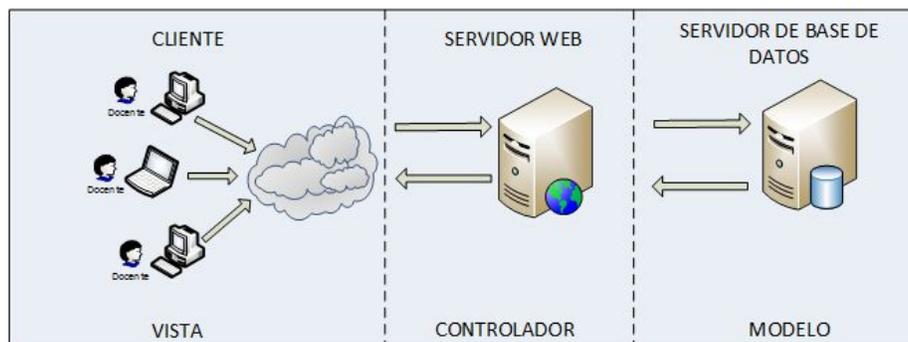


Ilustración 16: Arquitectura de la aplicación

Fuente: Elaboración propia

3.4 Seguridad

3.4.1 Seguridad a nivel de desarrollo

Para el desarrollo del sistema se utilizará el patrón Modelo Vista Controlador (MVC) para separar en 3 capas distintas la lógica del negocio, la interfaz o vista del usuario y el controlador, logrando con esto una mayor seguridad y control a tres niveles, además de una mejor facilidad de desarrollo y mantenimiento del sistema dado que de un lado se encuentra únicamente la representación visual que tendrá el usuario que no contiene ninguna información propia del negocio ni conexiones o demás ya que es el modelo que maneja todas estas gestiones siendo transparente para las demás capas. A nivel de controlador se gestiona todas las peticiones hechas por el usuario y se valida la información con el tipo de dato y estructura correspondiente que se espera manejando a este nivel la seguridad en cuanto a cambios malintencionados realizados desde el clientes o ataques forzados que intenten acceder al servidor y finalmente tenemos el modelo donde se encuentra toda la lógica e información del negocio, es aquí donde se gestiona todos los mecanismos de acceso, inserción, modificación y/o eliminación de datos de acuerdo

a reglas y procedimientos establecidos para conservar la integridad de la información.

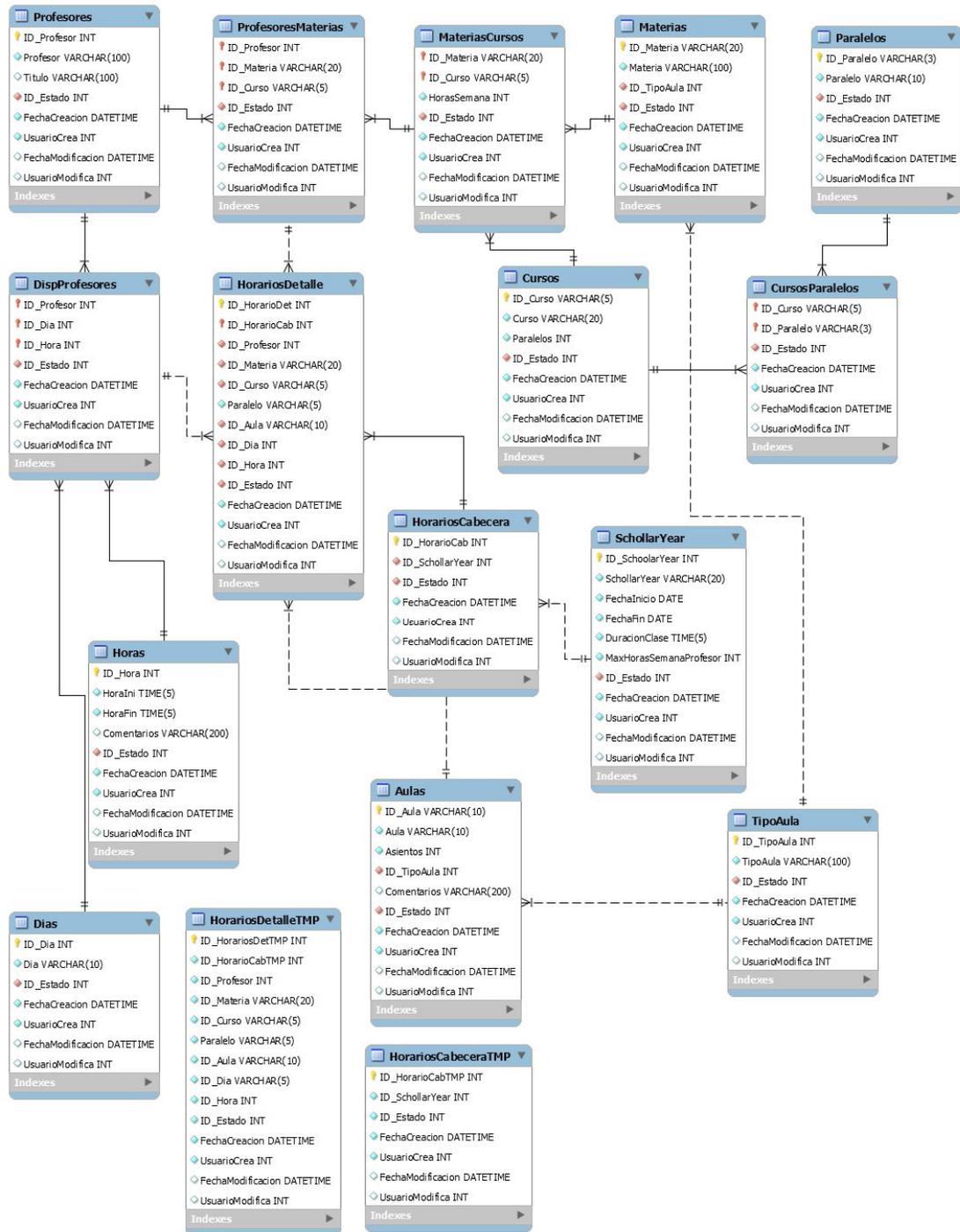
3.4.2 Seguridad en la base de datos

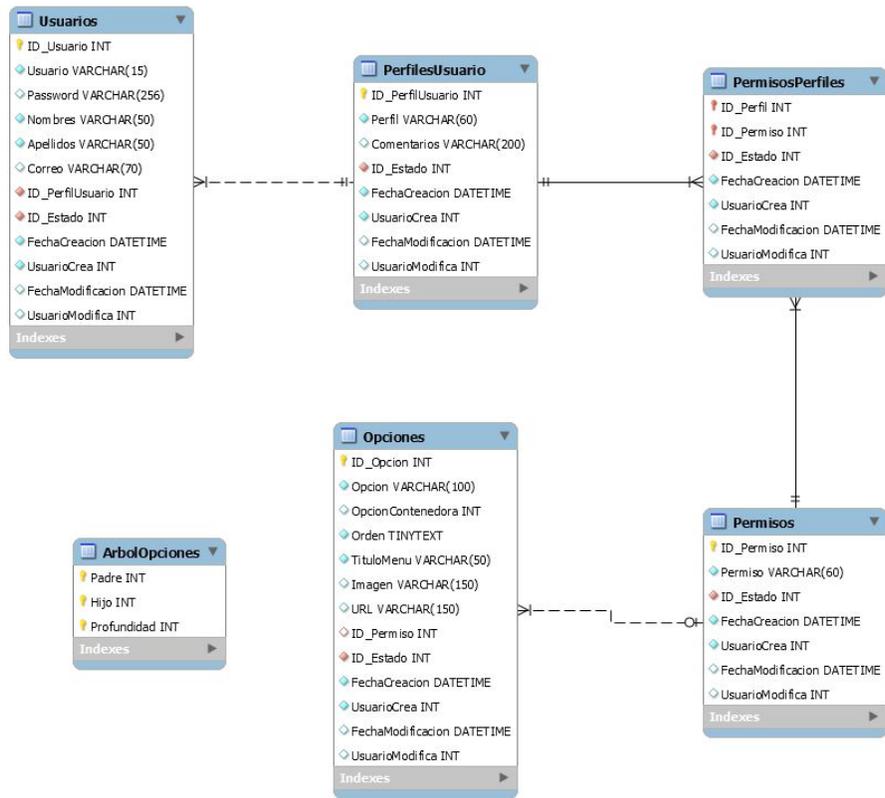
A nivel de base de datos se trabajará con un usuario sin privilegios de administrador que tendrá permisos únicamente sobre la base de datos del sistema. Las acciones permitidas por el usuario serán controladas de acuerdo a los permisos del tipo de perfil asignado al usuario que inicie sesión, el cual fue previamente registrado en la tabla correspondiente de usuarios utilizando el algoritmo de encriptación sha1 para almacenar la contraseña en el campo correspondiente, que de entre las principales razones de su elección es que es más seguro que MD5 debido a que su función de compresión basados en la técnica Hash es mucho más compleja, por lo que se lo considera más robusto y seguro comparado con otros algoritmos. La consulta, creación, actualización y/o eliminación de usuarios se las maneja mediante procedimientos almacenados en la base de datos para un mayor control del ingreso de datos al sistema y evitar la alteración de estructura de sentencias.

3.4.3 Seguridad en la web

Para controlar la seguridad a nivel web se utilizará e implementará certificados SSL para encriptar toda la información que viaje a través de internet, de esta forma cualquier petición POST o GET que realice el cliente desde la página web al servidor no podrá ser interpretada por algún usuario mal intencionado que intente revisar o interceptar el contenido de la acción pudiendo esta tener mayormente información sensible de algún usuario en particular del sistema que no debería poder ser consultada por terceros, logrando con esto una mayor seguridad en la confidencialidad de la información proporcionada para procesos y transacciones dentro del sistema, además de evitar que estas sean alteradas o manipuladas antes de llegar al servidor.

3.5 Diagrama Entidad Relación





3.6 Casos de uso

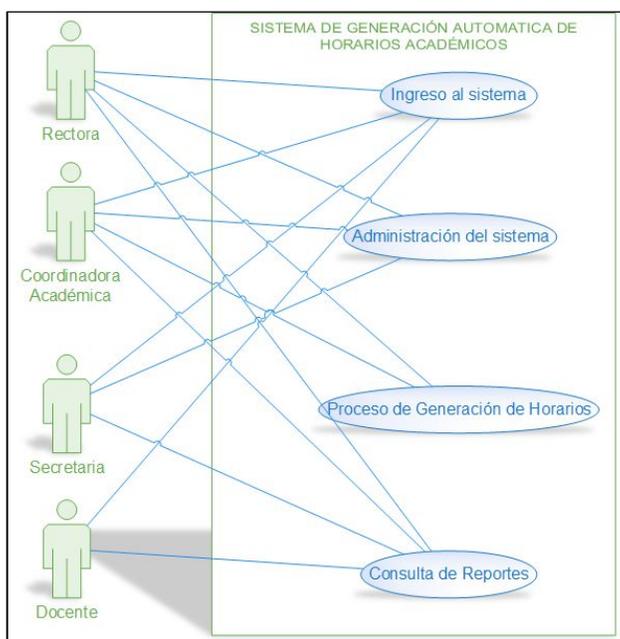


Ilustración 17: Casos de uso del Sistema de Generación de Horarios
Fuente: Elaboración propia

Descripción del caso de uso: Sistemas de Generación de Horarios	
Actor	Coordinadora académica, Secretaria, Docente, Rectora
Objetivo	El sistema
Descripción	Los actores pueden ingresar al sistema, administrar el sistema y generar horarios
Secuencia	Los actores ingresan al sistema. La coordinadora administra el sistema. Los actores generar los horarios. Los actores consultan reportes.

Tabla 7: Descripción del Casos de uso del Sistema de Generación de Horarios
Fuente: Elaboración propia

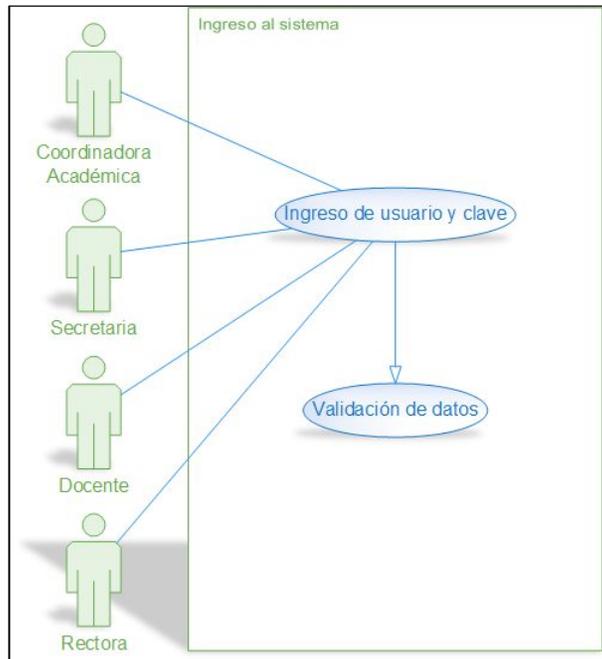


Ilustración 18: Casos de uso del ingreso al sistema
Fuente: Elaboración propia

Descripción del caso de uso: Ingreso al sistema	
Actor	Coordinadora académica, Secretaria, Docente, Rectora
Objetivo	Ingresar al sistema
Descripción	Los actores pueden ingresar al sistema
Secuencia	Los actores ingresan su usuario con su clave. El sistema valida que los datos ingresados sean correctos

Tabla 8: Descripción de los casos de uso del ingreso al sistema
Fuente: Elaboración propia

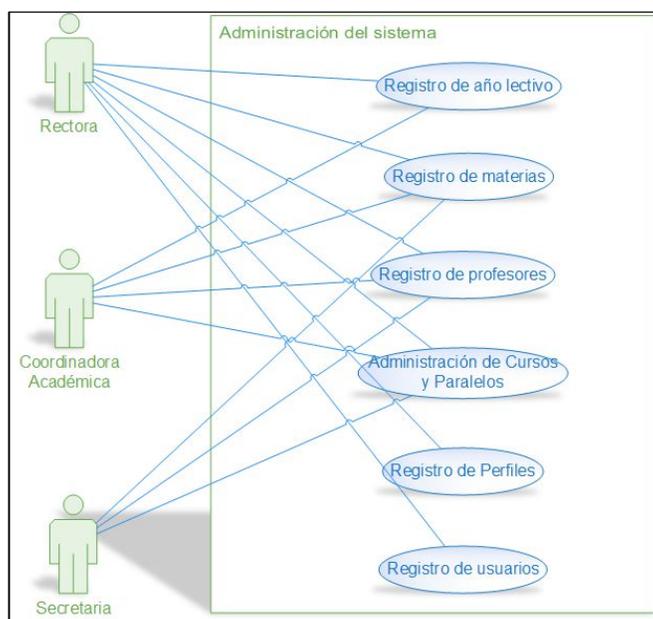


Ilustración 19: Casos de uso de administración del sistema
Fuente: Elaboración propia

Descripción del caso de uso: Administración del sistema	
Actor	Rectora, Coordinadora Académica y Secretaria
Objetivo	Administrar las opciones del sistema académico
Descripción	La rectora o la coordinadora académica gestionan la información necesaria para poder realizar la generación automática de los horarios. La secretaria solo tiene acceso a ciertas opciones del módulo
Secuencia	La rectora o la coordinadora académica registran el año lectivo. La rectora, la coordinadora o la secretaria registran las materias. La rectora, la coordinadora o la secretaria registran los profesores. La rectora, la coordinadora o la secretaria gestionan los cursos y paralelos. La rectora registra los perfiles. La rectora registra los usuarios

Tabla 9: Descripción de casos de uso de administración del sistema
Fuente: Elaboración propia

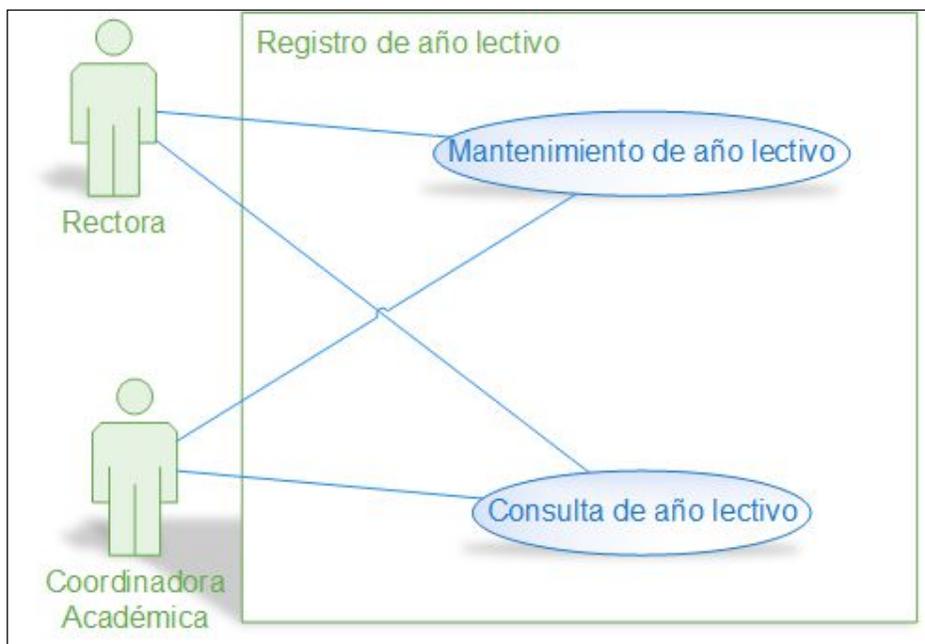


Ilustración 20: Casos de uso de registro de año lectivo

Fuente: Elaboración propia

Descripción del caso de uso: Registro de año lectivo	
Actor	Rectora, Coordinadora académica.
Objetivo	Registrar el año lectivo en curso
Descripción	La rectora o la coordinadora académica registra el año lectivo en curso
Secuencia	La rectora o la coordinadora académica pueden crea, modificar, eliminar el año lectivo. La rectora o la coordinadora académica pueden consultar los datos del año lectivo.

Tabla 10: Descripción de casos de uso de registro de año lectivo

Fuente: Elaboración propia

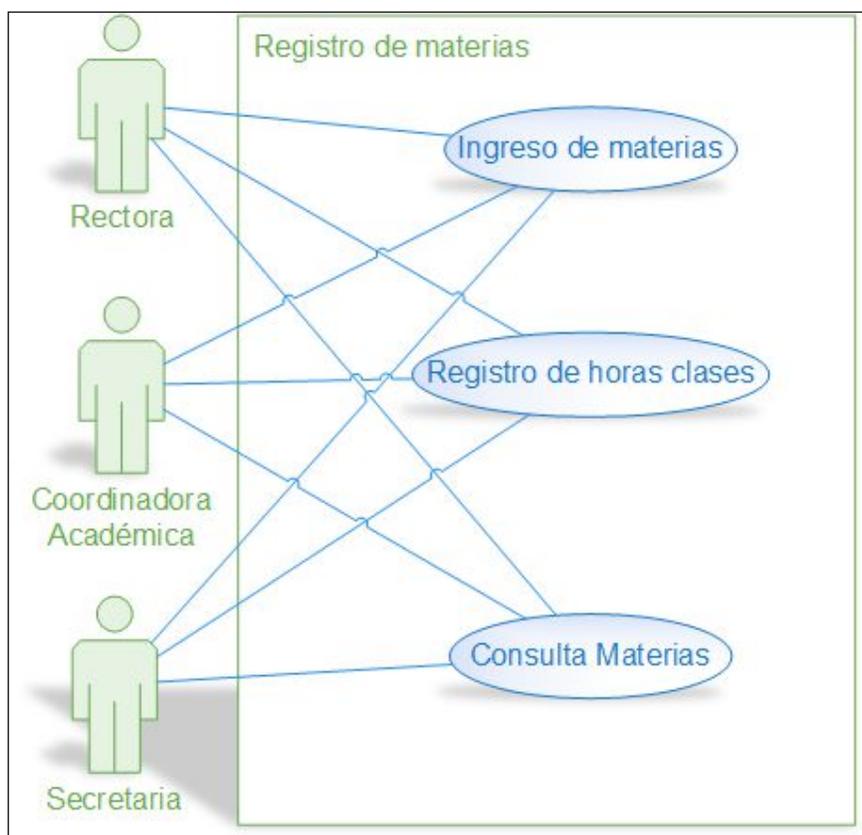


Ilustración 21: Casos de uso de Registro de materias
Fuente: Elaboración propia

Descripción del caso de uso: Registro de materias	
Actor	Rectora, Coordinadora académica, Secretaria
Objetivo	Registrar las materias del pensum académico
Descripción	Rectora, coordinadora académica o secretaria registra las materias que corresponden al pensum académico
Secuencia	La rectora, coordinadora académica o la secretaria ingresan las materias. La rectora, coordinadora académica o la secretaria registran las horas clase de las materias. La rectora, coordinadora académica o la secretaria consultan materias.

Tabla 11: Descripción de casos de uso de Registro de materias
Fuente: Elaboración propia

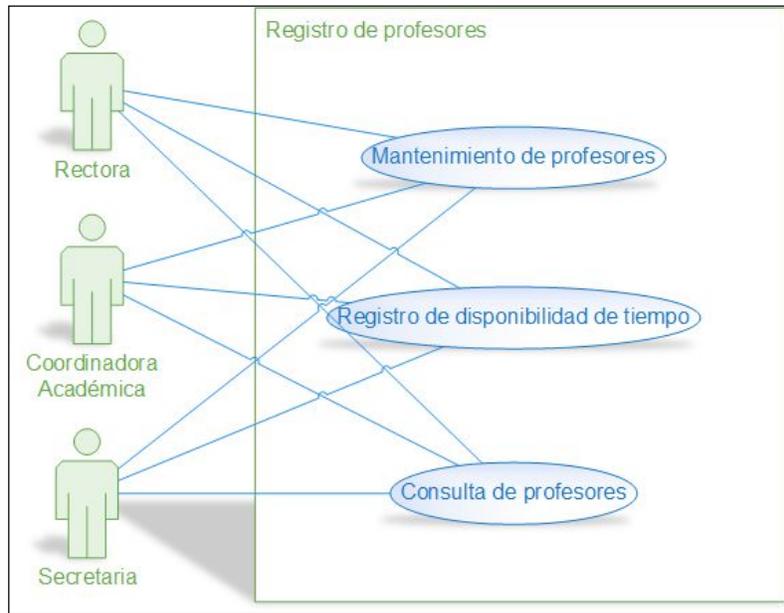


Ilustración 22: Casos de uso de registro de profesores
Fuente: Elaboración propia

Descripción del caso de uso: Registro de profesores	
Actor	Rectora, Coordinadora académica, Secretaria
Objetivo	Registrar los profesores
Descripción	La rectora, coordinadora académica o la secretaria registran a los profesores con los que contará el colegio durante el año lectivo
Secuencia	<p>La rectora, coordinadora académica o la secretaria ingresa, modifica o elimina a los profesores.</p> <p>La rectora, coordinadora académica o la secretaria registra la disponibilidad de tiempo de los profesores</p> <p>La rectora, coordinadora académica o la secretaria consultan los nombres de los profesores</p>

Tabla 12: Descripción de casos de uso de registro de profesores
Fuente: Elaboración propia

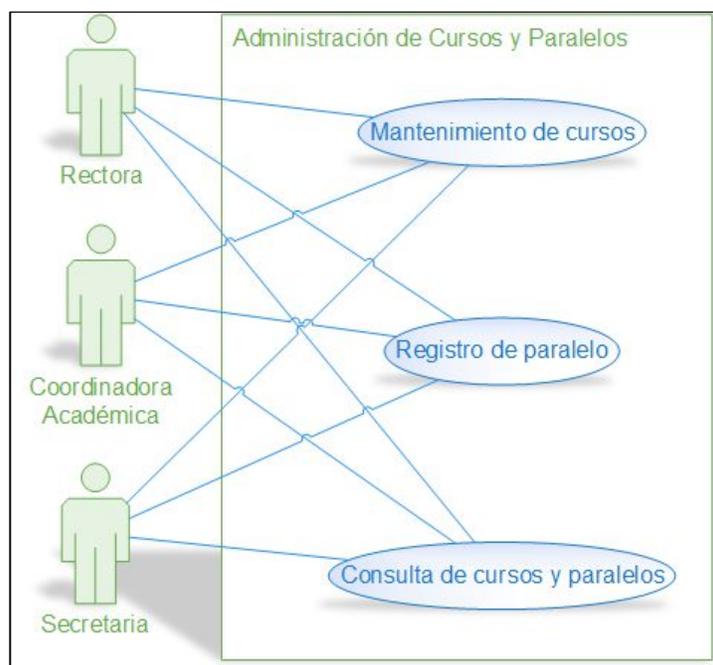


Ilustración 23: Casos de uso de administración de cursos y paralelos
Fuente: Elaboración propia

Descripción del caso de uso: Administración de cursos y paralelos	
Actor	Rectora, Coordinadora académica, Secretaria
Objetivo	Registrar los cursos y paralelos del colegio
Descripción	La rectora, coordinadora académica o la secretaria registran los cursos de ciclo básico y del bachillerato unificado
Secuencia	La rectora, coordinadora académica o la secretaria ingresa, modifica o elimina cursos. La rectora, coordinadora académica o la secretaria registra los paralelos necesarios. La rectora, coordinadora académica o la secretaria pueden consultar los cursos y paralelos creados.

Tabla 13: Descripción de casos de uso de administración de cursos y paralelos
Fuente: Elaboración propia

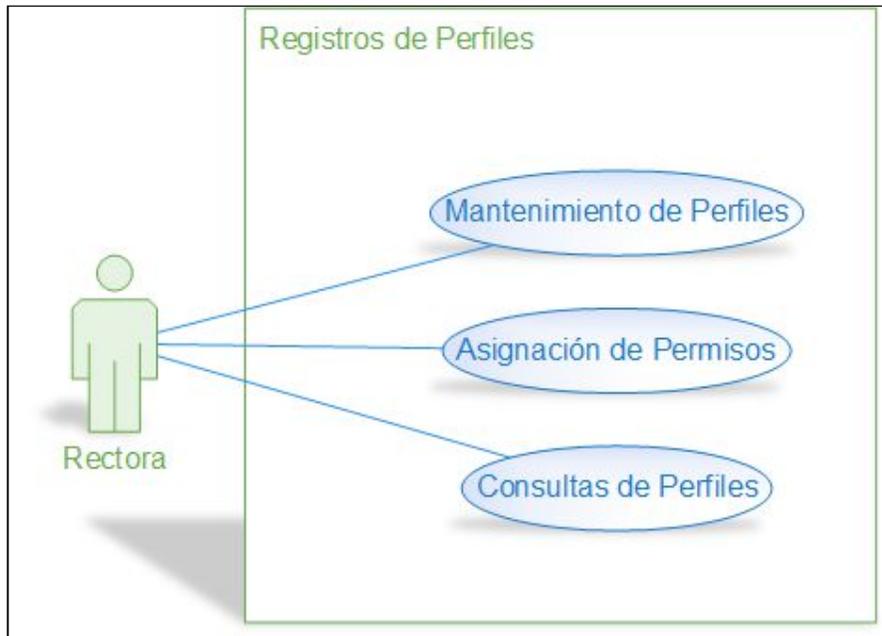


Ilustración 24: Casos de uso de registro de perfiles

Fuente: Elaboración propia

Descripción del caso de uso: Registro de perfiles	
Actor	Rectora
Objetivo	Registrar los perfiles del sistema
Descripción	La rectora registra los perfiles necesarios para el sistema
Secuencia	La rectora ingresa, modifica o elimina perfiles. La rectora asigna los permisos a los perfiles. La rectora puede consultar los perfiles creados.

Tabla 14: Descripción de casos de uso de registro de perfiles

Fuente: Elaboración propia

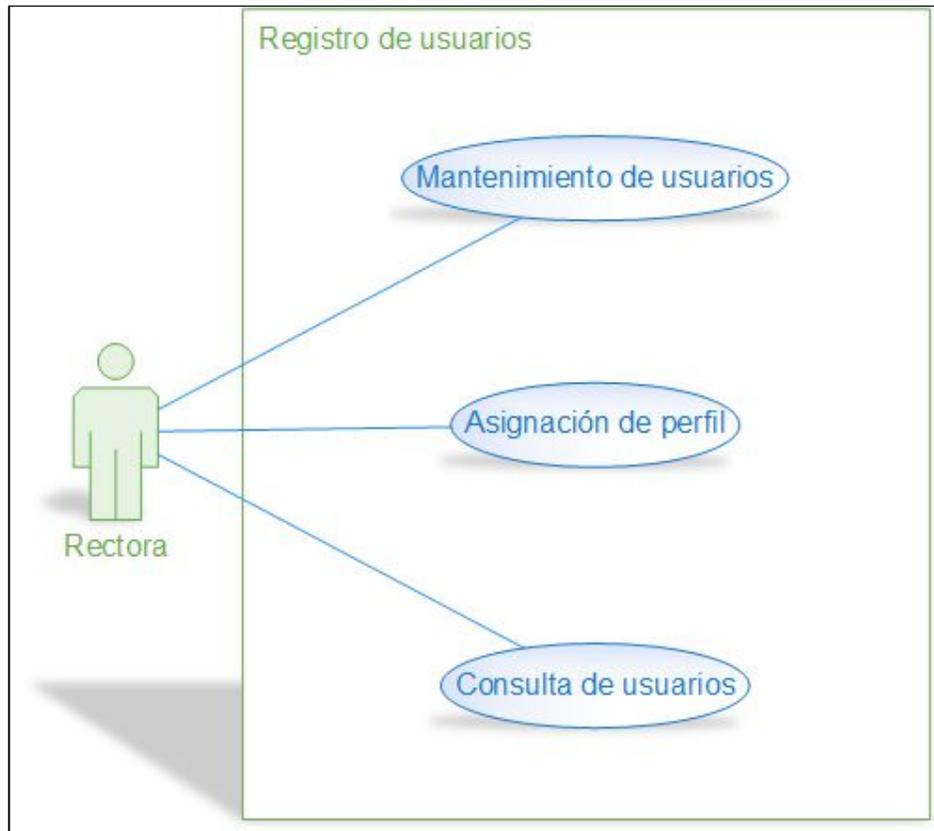


Ilustración 25: Casos de uso de registro de usuarios
Fuente: Elaboración propia

Descripción del caso de uso: Registro de usuarios	
Actor	Rectora
Objetivo	Mantenimiento de los usuarios de los docentes
Descripción	La rectora crea los usuarios de los profesores para el uso del sistema
Secuencia	La rectora crea los usuarios La rectora asigna el perfil de usuario La rectora consultan los usuarios registrados.

Tabla 15: Descripción de casos de uso de registro de usuarios
Fuente: Elaboración propia

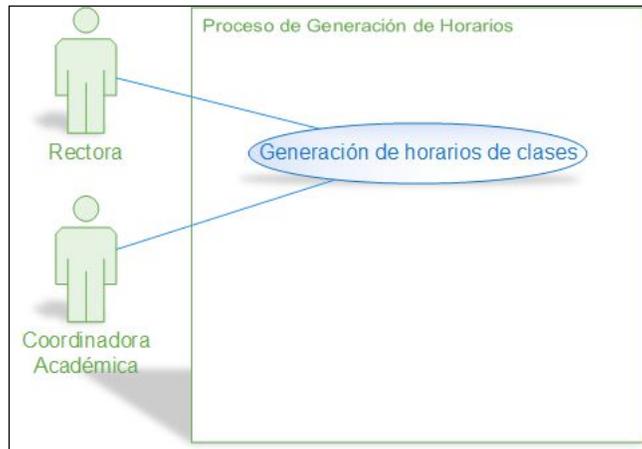


Ilustración 26: Casos de uso del proceso de generación de horarios
Fuente: Elaboración propia

Descripción del caso de uso: Proceso de Generación de horarios	
Actor	Rectora, Coordinadora académica
Objetivo	Realizar la generación automática de horarios de clases
Descripción	En este módulo está la opción de generación automática de horarios de clases.
Secuencia	La rectora genera los horarios automáticos. Para generar los horarios el proceso consulta materias, consulta profesor, consulta los cursos y paralelos y dependiendo de todas las variables anteriores generan el horario.

Tabla 16: Descripción del caso de uso del proceso de generación de horarios
Fuente: Elaboración propia

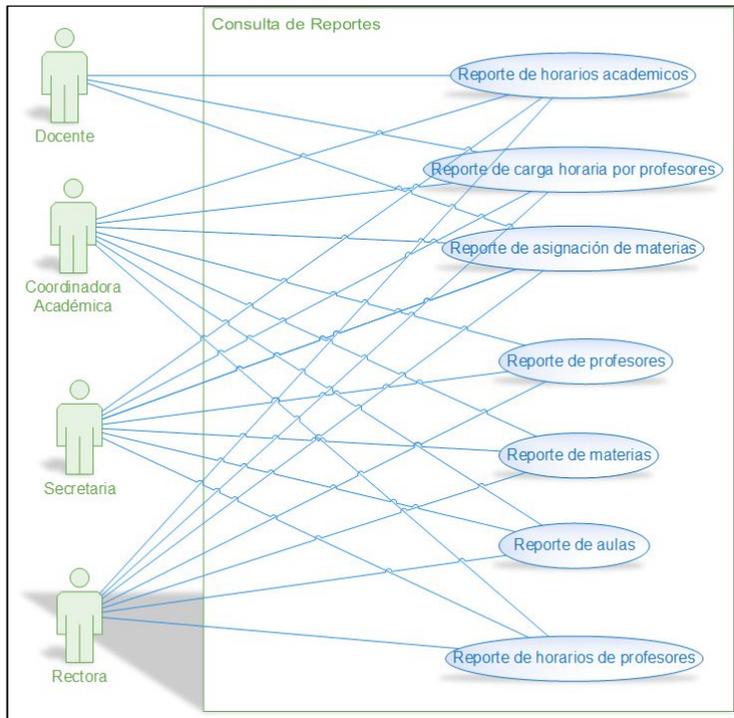


Ilustración 27: Casos de uso de consulta de Reportes
Fuente: Elaboración propia

Descripción del caso de uso: Consulta de reportes	
Actor	Rectora, Coordinadora académica, Docente, Secretaria.
Objetivo	Consultar los diferentes reportes generados por el sistema.
Descripción	En este módulo está la opción de consultar los diferentes reportes generados por el sistema.
Secuencia	La rectora, coordinadora académica, los docentes o la secretaria pueden consultar los horarios dependiendo de los permisos asignados.

Tabla 17: Descripción del caso de uso de consulta de Reportes
Fuente: Elaboración propia

3.7 Estudio de Factibilidad

Ya en conocimiento de las herramientas a utilizarse para el desarrollo de la página y el sistema web, es necesario realizar una evaluación de la factibilidad técnica y económica del proyecto analizándolos de la siguiente manera:

3.7.1 Estudio de Factibilidad Técnica

Para el desarrollo del sistema web, se han definido las siguientes herramientas teniendo como factor común el código libre y licencias sin costo:

Herramienta	Observaciones
Java/AJAX	Lenguaje de programación usado para el desarrollo
Eclipse Luna	IDE utilizado para el desarrollo
ZK Framework	Framework orientado al desarrollo web
MySQL	Motor de base de datos
JBoss	Servidor web para el desarrollo
Linux	Sistema Operativo Servidor

Tabla 18: Factibilidad técnica de las herramientas
Fuente: Elaboración propia

Recurso	Características
Espacio en Disco Duro recomendado	150 GB
Memoria RAM recomendada	16 GB
Plataforma recomendada	Linux

Tabla 19: Especificaciones técnicas recomendadas del host
Fuente: Elaboración propia

Recurso	Características
Espacio en Disco Duro recomendado	250 GB
Memoria RAM recomendada	16 GB
Plataforma recomendada	Windows
Sistema Operativo	Windows Server 2008 o superior
Procesador	Intel Core i5 similares o superior
Velocidad	2.50 GHz o superior

Tabla 20: Especificaciones técnicas recomendadas del servidor de servicios web
Fuente: Elaboración propia

3.7.2 Estudio de Factibilidad Económica

La unidad educativa “Federico González Suárez” no cuenta con un dominio propio ni con un hosting en donde almacenar el sistema, por lo que se adquirirá el servicio de hosting a la empresa FatCow (<http://www.fatcow.com/>) que es una empresa dedica a prestar servicios web, tales como dominios, hosting, administración de páginas web, entre otros beneficios, de los cuales se destaca es que cuentan también con certificados SSL necesarios para implementar un sistema web seguro. Por otro lado, el dominio web se lo adquirirá en (<http://www.nic.ec/>) ya que se necesita un dominio educativo local de un sitio seguro, confiable y con una larga trayectoria que garantice el servicio prestado.

Para realizar el análisis de la factibilidad económica, se toma en cuenta el hecho de que las herramientas utilizadas son de código abierto y licencias GPL, es decir, sin costo.

Herramienta	Costo por año	Observaciones
Eclipse Luna	\$ 0	Licencia GPL
ZK Framework	\$ 0	Licencia GPL
MySQL	\$ 0	Licencia GPL
Apache Tomcat	\$ 0	Licencia GPL
Linux	\$ 0	Licencia GPL
Dominio	\$ 40	Adquirido en NIC.ec
Hosting	\$ 49	El servicio que ofrece FatCow
Certificado SSL	\$ 40	El servicio que ofrece FatCow
Licencia API o aplicativo GAMS	\$600	Licencia para aplicativo matemático GAMS

Tabla 21: Factibilidad económica de las herramientas

Fuente: Elaboración propia

CAPITULO IV: CONCLUSIONES Y RECOMENDACIONES

4.1 Conclusiones

De acuerdo al análisis de datos realizados por medio de las encuestas a los docentes se concluye que el presente trabajo de titulación es viable, porque de acuerdo a las estadísticas obtenidas se puede observar que el 95% de los profesores de la unidad educativa muestran mucho interés y aceptación al desarrollo e implementación de este sistema ya que para ellos puede proyectar a mejoras en la distribución de horarios tomando en consideración primordial la disponibilidad del profesor de acuerdo a la carga de materias que pueden impartir.

Adicional a esto, en la entrevista realizada a la rectora de la Unidad Educativa Federico González Suarez, el tiempo estimado de duración de la generación de horarios manual es de 8 horas, lo que no solo conlleva de la dedicación completa de los recursos asignados, sino que interfiere con éste en las demás tareas asignadas pendientes a realizar y por ende decrece el nivel de productividad general de la unidad educativa en cuestión, en contraste con la implementación del sistema web de generación automático de horarios académicos, el tiempo de elaboración disminuirá a aproximadamente 1 hora, logrando así una optimización de tiempo y recurso de un 87.5%,

4.2 Recomendaciones

Debido al constante mantenimiento de información sensible de la unidad educativa como para mantener su integridad, es importante para el correcto funcionamiento del sistema, la gestión de copias de seguridad del sitio como de la base de datos del sistema web mediante respaldos automatizados programados en un determinados tiempo, y la posterior revisión de los mismos con una persona encargada de realizar auditoria a la transaccionalidad de la información y probar que todos los respaldos generados funcionen correctamente y mantengan una nomenclatura previamente definida que ayude a identificar por fechas por cualquier inconveniente que se pueda presentar en un futuro.

Para trabajos futuros se recomienda la ampliación del sistema implementado para con dispositivos móviles y de esa manera ampliar los tipos de dispositivos mediante los cuales se tendrá acceso al sistema.

Por último, se recomienda implementar un módulo de envío de reportes a correos electrónicos de las autoridades del plantel y a los mails de los docentes para poder ampliar los medios de información acerca de novedades del mismo sistema.

5. Bibliografía

- Aguilar, J., Rivas, F., & Cerrada, M. (2014). Diseño de un Medio de Gestión de Servicios para Sistemas Multiagentes. *Universidad de los Andes, Venezuela.*, 1-9.
- Álvarez Monier, E., & Ávila Martínez, I. (2004). Algoritmo para hacer algoritmos. *E-brary*.
- Barroso Osuna, J., & Cabero-Almenara, J. (2013). Replanteando el e-learning: hacia el e-learning 2.0. *Campus Virtuales*, 2(2), 76-87.
- Durañona, S. C., Roche, M. del C., & Sarría Pérez, C. A. (2014). Base de datos para la automatización de un registro de usuarios. *Revista Cubana de Tecnología de la Salud*, 5(2), 1-5.
- Escalona, L. S. B., & Inclán, I. L. V. (2012). Funciones resúmenes o hash. *Revista Telemática*, 10(1). Recuperado a partir de <http://revistatelematica.cujae.edu.cu/index.php/tele/article/view/52>
- Eslava Muñoz, V. J. (2013). *El nuevo php: conceptos avanzados*. s.l.: Bubok.
- Funes, A., & Dasso, A. (2014). Evaluación de frameworks para aplicaciones web. Presentado en XVI Workshop de Investigadores en Ciencias de la Computación. Recuperado a partir de <http://hdl.handle.net/10915/41581>
- Gaona, P. A., Montenegro Marín, C. E., & Wiesner Gonzalez, H. (2014). Hacia una Propuesta de Mecanismos para la Autenticidad de Objetos de Aprendizaje en Plataformas LCMS. *Ingeniería*, 19(1). <http://doi.org/10.14483/udistrital.jour.reving.2014.1.a03>
- García Martín, J., & García Sánchez, J. N. (2013). Conocimiento diferencial que tienen los jóvenes españoles sobre la web 2.0. *International Journal of Developmental and Educational Psychology: INFAD. Revista de Psicología*, 2(1), 689-694.
- González, Y. D., & Romero, Y. F. (2012). Patrón Modelo-Vista-Controlador. *Revista Telemática*, 11(1), 47-57.
- González Menéndez, J. A. (2015). *Utilización de las bases de datos, relaciones en el sistema de gestión y almacenamiento de datos*. Madrid: Paraninfo.
- Guerrero Ibáñez, J. A., & Barba Martí, A. (2008). Un algoritmo de selección de celda de acceso para entornos de comunicaciones móviles de próxima generación. *IEEE LATIN AMERICA TRANSACTIONS*, 6(2), 194-200.
- Guez, G. R. (2012). *Software propietario versus software libre: oportunidades y retos*. Place of publication not identified: Editorial Acad Mica Espa.
- Hernández Sampieri, R., Fernández Collado, C., & Baptista Lucio, P. (2006). *Metodología de la investigación*. México: McGraw Hill.
- Martínez Martínez, A., & Cegarra Navarro, J. G. (2014). *Gestión por procesos de negocio: organización horizontal*. Madrid: Ecobook.
- Mas Ruiz, F. J. (2011). *Temas de investigación comercial*. San Vicente [del Raspeig: Editorial Club Universitario.

- Méndez Rojas, O. E. (2015). *Diseño de base de datos para el análisis de rendimientos de mano de obra*. TEC, COSTA RICA. Recuperado a partir de <http://repositoriotec.tec.ac.cr/handle/2238/6080>
- Montoya, C. E. G., Uribe, C. A. C., & Rodríguez, L. E. S. (2013). Seguridad en la configuración del servidor web Apache. *INGE CUC*, 9(2), 31-38.
- Moreno, J. Z. (2015). *Ciberdiccionario: Conceptos de ciberseguridad en lenguaje entendible*. Javier Zubieta.
- Moyota Chafra, J. C. (2015). *Análisis de Framework Opensource en el desarrollo de una aplicación web para la gestión de reportes estadísticos en el sistema de agua potable de la Provincia de Chimborazo*. UNACH, CHIMBORAZO. Recuperado a partir de <http://dspace.unach.edu.ec/handle/51000/725>
- Pilacúan Tepud, J. A. (2014). *Sistema web para el control de producción y tiempo perdido en la planta de pintura (GM)*. EPN, QUITO. Recuperado a partir de http://biblioteca.epn.edu.ec/cgi-bin/koha/opac-detail.pl?biblionumber=24790&shelfbrowse_itemnumber=33342
- Pousa, A., Sanz, V. M., & De Giusti, A. E. (2015). Comparación de rendimiento de algoritmos de cómputo intensivo y de acceso intensivo a memoria sobre arquitecturas multicore. Presentado en XXI Congreso Argentino de Ciencias de la Computación (Junín, 2015). Recuperado a partir de <http://hdl.handle.net/10915/50145>
- Poveda Paez, R. A., Herrera Vaca, P. I., & Torres Camposano, M. A. (2016, febrero 1). *Desarrollo e implementación de una aplicación web interactiva para donaciones, registros de voluntarios, campañas y publicaciones en el Banco de Alimentos Diakonia de Guayaquil*. Recuperado a partir de <http://www.dspace.espol.edu.ec/handle/123456789/31845>
- Ramos Martín, A. (2014). *Aplicaciones Web*. Madrid: Paraninfo.
- Roselló, R. A., & Núñez, M. M. (2013). Aplicaciones digitales online para la gestión de recursos docentes audiovisuales: el modelo del Labcom UJI. *Estudios sobre el Mensaje Periodístico*, 19(0), 55-63. http://doi.org/10.5209/rev_ESMP.2013.v19.42011
- Ruiz del Castillo, J. C., & Huertas, P. L. (2014). La web 2.0 en el entorno empresarial: Socializando las Pymes. *Pixel-Bit: Revista de medios y educación*, (45), 65-77.
- Salazar, Z., Esteban, J., & Trueba Espinosa, A. (2015). Algoritmo computacional de mezcla dinámica de cliques para medir la resonancia de individuos en redes sociales. *Acta universitaria*, 25(2), 28-39. <http://doi.org/10.15174/au.2015.733>
- Solinas, M., & Antonelli, L. (2013). Software evolution and design patterns. *IEEE Latin America Transactions*, 11(1), 347-352. <http://doi.org/10.1109/TLA.2013.6502828>
- Strauss, A. L., Corbin, J., & Zimmerman, E. (2002). *Bases de la investigación cualitativa: técnicas y procedimientos para desarrollar la teoría fundamentada*. Medellín (Colombia): Universidad de Antioquía.
- Talledo San Miguel, J. (2015). *Implantación de aplicaciones web en entornos internet, intranet y extranet*. Madrid: Paraninfo.

Vasquez, J. (2014). *Libro científico: Investigaciones en tecnologías de información informática y computación*. Palibrio.

DB-Engines. (Julio de 2016). *DB-Engines Ranking of Relational DBMS*. Obtenido de <http://db-engines.com>

Krause, M. (1995). La investigación cualitativa: un campo de posibilidades y desafíos. *Revista Temas de educación* 7, 19-40.

6. Anexos

6.1 Anexo 1: Entrevista a Rectora de la Unidad Educativa Federico González Suarez del Cantón Durán

Nombre: Lic. Carmen Ojeda

Fecha: 24 de Junio del 2016

Hora: 09:00 am

1. ¿Actualmente cuentan con un sistema académico?

Si, manejamos uno ya hace 5 a 6 años.

2. ¿El sistema académico cuenta con la opción de generación de horarios de clase?

No, para la generación de horarios de clase utilizo otro software específicamente solo para eso.

3. ¿Cuánto tiempo le lleva elaborar el horario de clases para el año lectivo?

Sin levantarme, 8 horas.

4. ¿Quiénes participan en la elaboración de los horarios de clases para el año lectivo?

Normalmente el vicerrector, las inspectora y yo.

5. ¿Cuántas materias básicas tiene el pensum académico?

De toda la institución, desde octavo básico a tercero de bachillerato son 50 materias.

6. ¿Cuáles son las materias básicas?

Lengua y literatura, matemáticas, estudios sociales, ciencias naturales, o sea esas son las áreas básicas, de ahí en el bachillerato, por ejemplo, de estudios sociales se divide en historia y geografía, educación para la ciudadanía y en el área de ciencias naturales se divide que física, química, biología.

7. ¿Cuántas materias de especialidad tiene el pensum académico?

El bachillerato general unificado, obviamente no es especialidad, las especialidades son en contabilidad y en informática y creo que son alrededor de 10 materias por cada especialidad entre primero, segundo, y tercero del bachillerato.

8. ¿Cuáles son las materias de especialidad?

No me las sé de memoria, por ejemplo en informática es programación en lenguajes estructurados, es una de ellas que es la más fuerte, y en el área de contable es contabilidad y tesorería, o sea, esas son las más fuertes que manejan en este momento, de ahí el resto son materias de especialidad pero los nombres todos no me los sé de memoria.

9. Mencione las jornadas académicas con las que cuenta la unidad educativa

Ya este año empezamos a funcionar sólo con matutina, anteriormente teníamos las tres secciones, matutina, vespertina y nocturna, pero la vespertina se suspendió hace unos 4 a 5 años y la nocturna ya se suspendió este año.

10. ¿Cuántas horas clase hay en la jornada estudiantil?

8 horas hasta la educación básica y 9 horas el bachillerato.

11. ¿Cuántas materias se dan por curso?

En promedio, la educación básica son 12 materias, el bachillerato algunos cursos tienen 14 y otros 15.

12. ¿Hasta cuántas horas clase puede tener una materia?

Eso depende del pensum que te da el ministerio, ahí, por ejemplo, programación tiene 7 horas a la semana, matemáticas y lenguas en el ciclo básico son 6 horas a la semana. O sea, ya el ministerio te da la carga horaria y ya con eso tú tienes que manejarlo, pero también te da libertad a las instituciones para la gestión de horas dentro de la jornada que tú le quieras poner de acuerdo a las necesidades de la institución.

13. ¿Cómo realiza la distribución de horas clase por materia?

Ellos (el ministerio) te dicen cuántas horas tienes que dar, pero nosotros distribuimos cuántas horas se dan, 2 horas diarias, 1 una hora diaria, eso depende de la institución como lo maneja.

La distribución de horas clase por materia se realiza considerando que no sea agotador para el maestro ni para el estudiante, cualquiera se aburre de verse la cara 8 horas diarias entonces se trata de que sea por lo menos una hora diaria y que si tienen 2 horas diarias, las horas estén juntas para no perder el hilo.

14. ¿Con cuántos profesores cuenta actualmente el colegio?

21 profesores.

15. ¿Hasta cuántas materias puede dar un profesor?

No es por materia, es por la carga horaria, un maestro en promedio tiene 34 horas a la semana, en promedio, porque unos tienen 36, otros 35, otros 33, otros 32, pero el promedio es 34 horas de clase a la semana, en la cantidad de materias que sean, en la cantidad de cursos que sean, pero son 34 horas el promedio.

16. ¿Cómo realiza la distribución de materias por profesor?

El pensum me dice, por ejemplo, que tengo que dar 5 horas de matemáticas y 5 horas de física, entonces yo al profesor, tengo 3 cursos, en total son 15 horas de clase, entonces tengo un profesor que me puede dar esas 6 materias, porque en total son 30 más otra materia que sean solo 4 horas, 34, o sea ya dependen de cuantas horas se da por curso yo le voy poniendo al profesor, le pongo, le pongo, le pongo hasta sumar cuantas están.

17. ¿Los horarios académicos generados son publicados en el internet para que sean consultados por los profesores?

No.

6.2 Anexo 2: Formato de encuesta a los Docentes de la Unidad Educativa Federico González Suarez del Cantón Durán

Estimado(a):

La siguiente encuesta es de carácter investigativo y se realiza con la finalidad de obtener el punto de vista de cada uno de los docentes, respecto a la implementación de un sistema web para generar horarios académicos para la Unidad Educativa Federico González Suarez del Cantón Durán.

Por favor marcar sus respuestas con una (X):

1. **Sexo:**

- Masculino Femenino

2. **Edad:**

- 18 - 23 años 36 - 41 años
 24 - 29 años 42 - 47 años
 30 - 35 años Más de 48 años

3. **¿La institución arma su horario de clases conforme a su disponibilidad?**

- Si No

4. **¿De qué manera le dan a conocer su horario de clase?**

- Mail Presencial Llamada Telefónica

5. **¿Quién le da a conocer el horario académico?**

- Secretaria Coordinador(a)
 Vicerrector(a) Inspector(a)

6. **¿Cuántas materias tiene asignadas actualmente?**

- 1 – 3 7 – 9
 4 – 6 10 o más

7. **¿Es dirigente de algún curso?**

- Si No

8. **¿Está de acuerdo con la implementación de un sistema informático que genere los horarios de clase de forma automática, considerando su horario de disponibilidad?**

- Si No

9. **¿A través del internet, le gustaría tener acceso al sistema informático para conocer su horario de clase desde cualquier sitio?**
- Si No
10. **¿Le informan con anticipación su horario de clase?**
- Si No
11. **¿Con que tiempo de anticipación le informan su horario de clases?**
- 0 días 1 – 7 días Más de 8 días
12. **¿Qué tan dispuesto estaría ud. en utilizar el sistema informático para consultar sus horarios académicos por Internet?**
- 1 No Dispuesto 4 Dispuesto
 2 Poco Dispuesto 5 Muy Dispuesto
 3 Indiferente
13. **¿Por qué no utilizaría el sistema informático para consultar los horarios académicos?**
- Acceso a Internet Seguridad
 No dispone de medios Otros
14. **¿Por qué si utilizaría el sistema informático para consultar los horarios académicos?**
- Ahorro de Tiempo Información en Línea
 Uso de Tecnología Otros

6.3 Anexo 3: Manual de Usuario

Contenido:

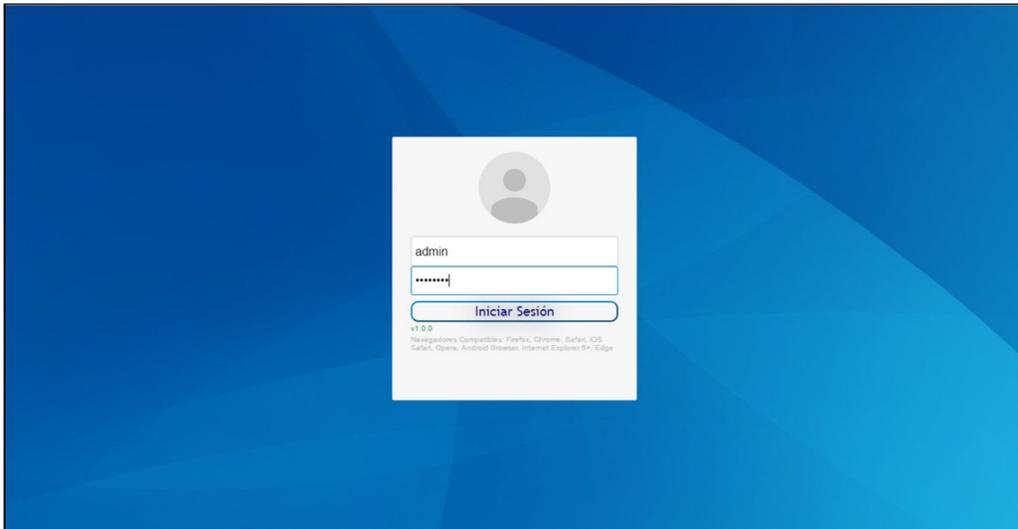
- 1. Introducción al usuario**
- 2. Ingreso al sistema**
- 3. Menú**
- 4. Mantenimiento**
 - 4.1 Año Lectivo**
 - 4.2 Cursos**
 - 4.3 Materias**
 - 4.4 Aulas**
 - 4.5 Tipos de aula**
 - 4.6 Profesores**
 - 4.7 Usuarios**
- 5 Procesos**
 - 5.1 Gestor de Horarios Académicos**
- 6 Reportes**

1. Introducción al usuario

En el siguiente documento se especifica con detalle la operatividad del Sistema Gestor Automático de Horarios Académicos conforme a la gestión y elaboración de horarios de clases para el año lectivo.

2. Ingreso al sistema

Para poder acceder al sistema se deberá ingresar el usuario y contraseña respectivo, los cuales ya deberán haber sido creados previamente en el módulo de usuarios y posterior dar click en el botón *Iniciar Sesión*.



Una vez ingresado al sistema se mostrará la pantalla principal del sistema en donde se mostrará el menú principal del mismo y el escudo de la unidad educativa



3. Menú

El sistema cuenta con los módulos respectivos para la administración del año lectivo, cursos, materias, aulas, tipos de aulas, profesores y usuarios. Dentro del módulo de usuarios se da las opciones para administrar los datos y perfiles de cada usuario, así como también se asignan los permisos.

Dentro del sistema se encuentra el módulo principal de la generación automática del horario académico del año lectivo, el cual podrá ser generado una vez se cumplan con los registros totales de la información necesaria en los módulos anteriores. Para control acerca del sistema se emplea un módulo de reportes en donde se emitirán reportes sobre materias y profesores.

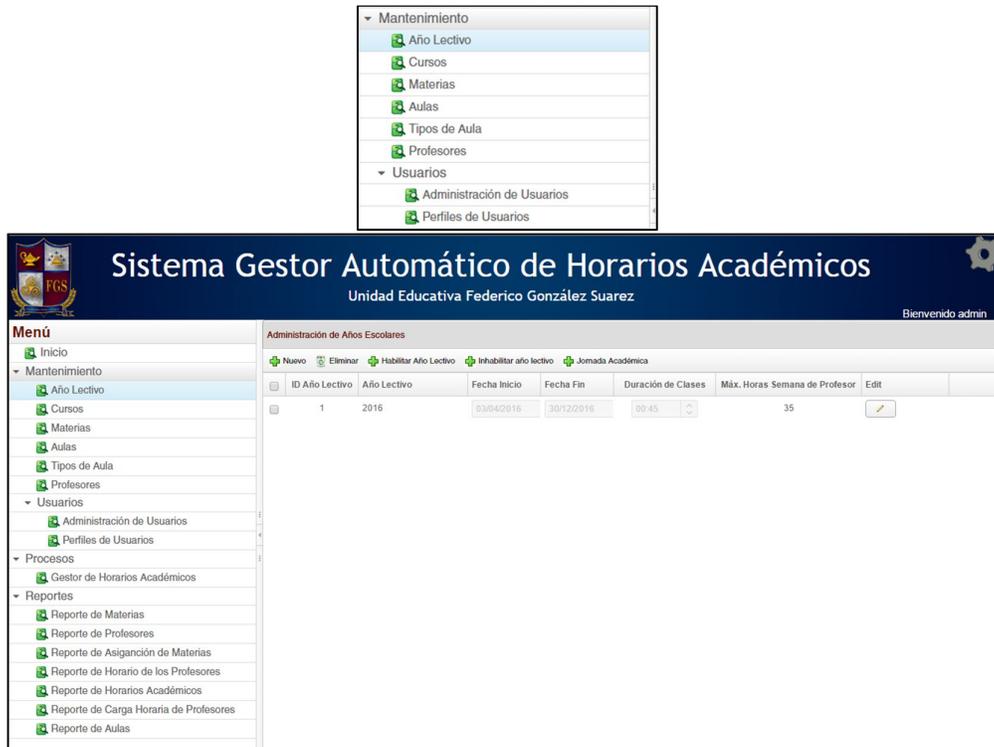


4. Mantenimiento

Dentro del módulo de mantenimiento se da al usuario la opción de crear, modificar y eliminar cada una de las variables necesarias para poder generar el horario académico.

4.1 Año Lectivo

Para ingresar a la opción de mantenimiento de Año Lectivo se debe de dar click en la opción del menú con el mismo nombre dentro del módulo de mantenimiento y se abrirá la siguiente pantalla:



Dentro del mantenimiento del año lectivo se encuentran los siguientes botones:



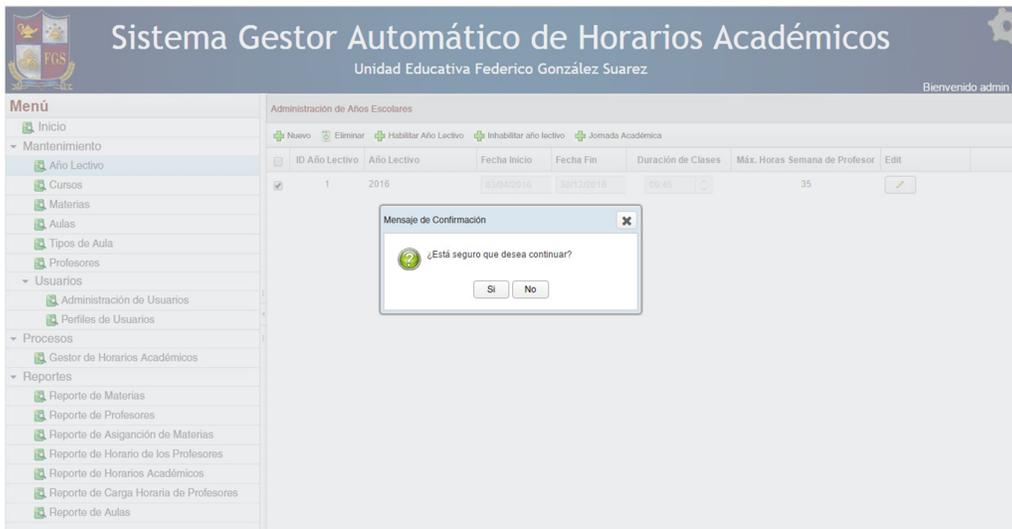
- Nuevo – Permite al usuario crear un nuevo año lectivo desplegando la siguiente pantalla:



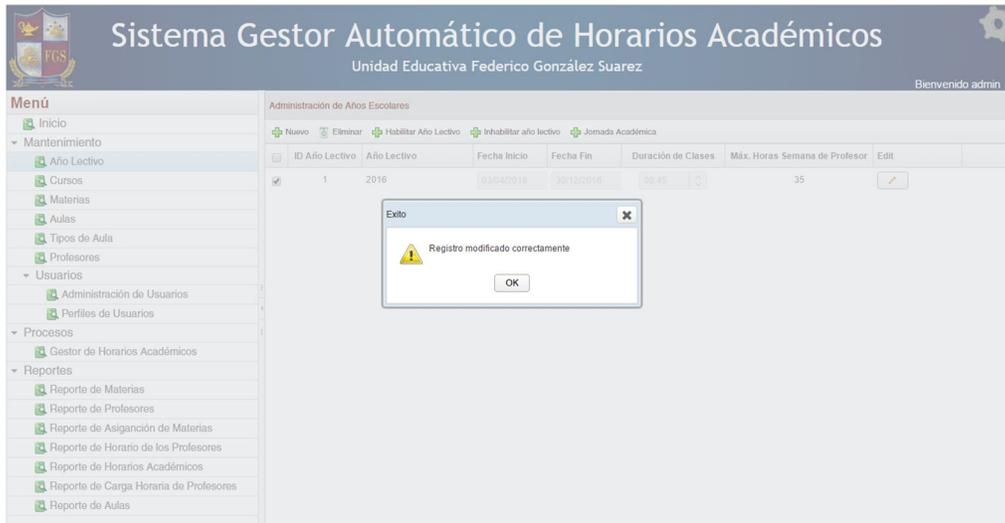
- Editar – Permite al usuario editar el año lectivo creado presionando en el botón  habilitando los campos de edición del año lectivo y mostrando los botones de  guardar o  cancelar los cambios mostrando la pantalla de la siguiente forma:



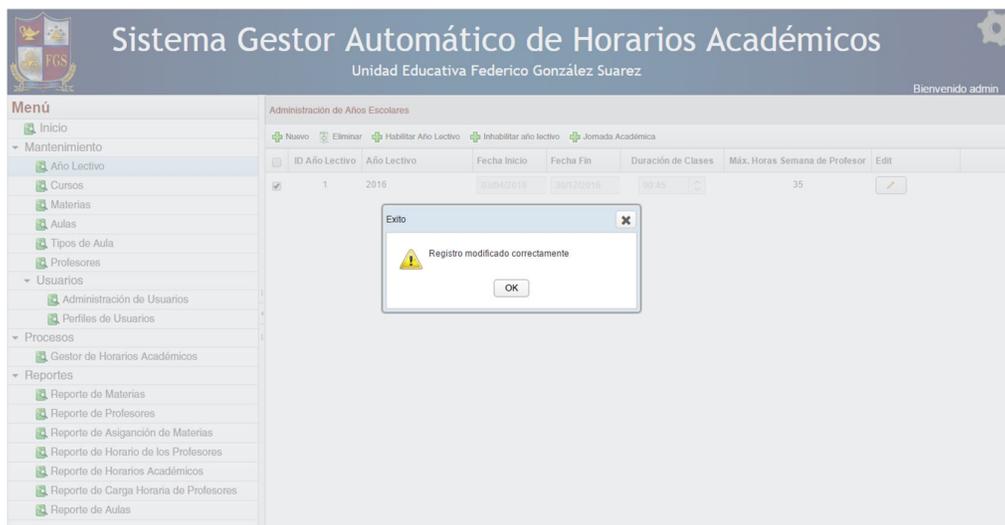
- Eliminar – Permite al usuario eliminar un año lectivo seleccionado de la lista mostrando el siguiente mensaje de confirmación:



- Habilitar Año Lectivo – Permite al usuario poner en vigencia y funcionamiento un año lectivo seleccionado de la lista mostrando el siguiente mensaje:



- Inhabilitar Año Lectivo – Permite al usuario inhabilitar un año lectivo seleccionado de la lista mostrando el siguiente mensaje:



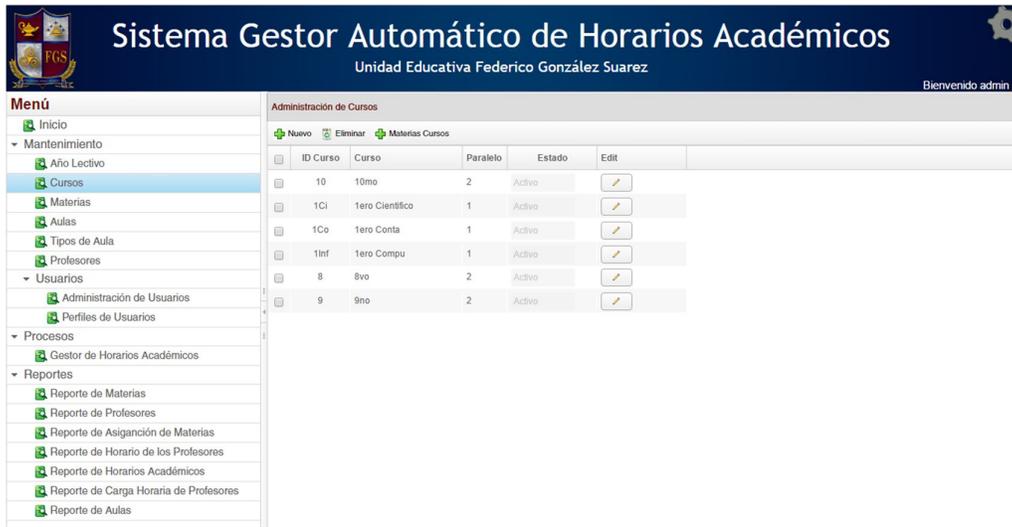
- Jornada Académica – Permite al usuario configurar la jornada académica del año lectivo mostrando la siguiente pantalla:



4.2 Cursos

Para ingresar a la opción de mantenimiento de cursos se debe de dar click en la opción del menú con el mismo nombre dentro del módulo de mantenimiento y se abrirá la siguiente pantalla:

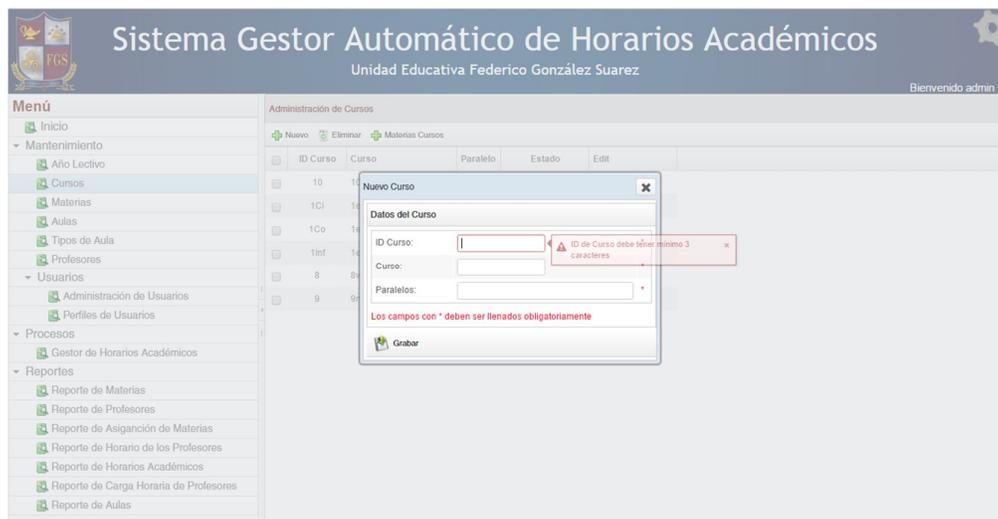




Dentro del mantenimiento de cursos se encuentran los siguientes botones:



- Nuevo – Permite al usuario crear un nuevo curso desplegando la siguiente pantalla:



- Editar – Permite al usuario editar el curso creado presionando en el botón habilitando los campos de edición del curso y mostrando los botones de guardar o cancelar los cambios mostrando la pantalla de la siguiente forma:

Sistema Gestor Automático de Horarios Académicos
Unidad Educativa Federico González Suarez

Bienvenido admin

Menú

- Inicio
- Mantenimiento
 - Año Lectivo
 - Cursos
 - Materias
 - Aulas
 - Tipos de Aula
 - Profesores
 - Usuarios
 - Administración de Usuarios
 - Perfiles de Usuarios
- Procesos
 - Gestor de Horarios Académicos
- Reportes
 - Reporte de Materias
 - Reporte de Profesores
 - Reporte de Asignación de Materias
 - Reporte de Horario de los Profesores
 - Reporte de Horarios Académicos
 - Reporte de Carga Horaria de Profesores
 - Reporte de Aulas

Administración de Cursos

Nuevo Eliminar Materias Cursos

ID Curso	Curso	Paralelo	Estado	Edit
10	10mo	2	Activo	<input type="checkbox"/> <input type="checkbox"/>
1Ci	1ero Científico	1	Activo	<input type="checkbox"/> <input type="checkbox"/>
1Co	1ero Conta	1	Activo	<input type="checkbox"/> <input type="checkbox"/>
1Inf	1ero Compu	1	Activo	<input type="checkbox"/> <input type="checkbox"/>
8	8vo	2	Activo	<input type="checkbox"/> <input type="checkbox"/>
9	9no	2	Activo	<input type="checkbox"/> <input type="checkbox"/>

- Eliminar – Permite al usuario eliminar un curso seleccionado de la lista mostrando el siguiente mensaje de confirmación:

Sistema Gestor Automático de Horarios Académicos
Unidad Educativa Federico González Suarez

Bienvenido admin

Menú

- Inicio
- Mantenimiento
 - Año Lectivo
 - Cursos
 - Materias
 - Aulas
 - Tipos de Aula
 - Profesores
 - Usuarios
 - Administración de Usuarios
 - Perfiles de Usuarios
- Procesos
 - Gestor de Horarios Académicos
- Reportes
 - Reporte de Materias
 - Reporte de Profesores
 - Reporte de Asignación de Materias
 - Reporte de Horario de los Profesores
 - Reporte de Horarios Académicos
 - Reporte de Carga Horaria de Profesores
 - Reporte de Aulas

Administración de Cursos

Nuevo Eliminar Materias Cursos

ID Curso	Curso	Paralelo	Estado	Edit	
<input checked="" type="checkbox"/>	10	10mo	2	Activo	<input type="checkbox"/> <input type="checkbox"/>
<input type="checkbox"/>	1Ci	1ero C			<input type="checkbox"/> <input type="checkbox"/>
<input type="checkbox"/>	1Co	1ero C			<input type="checkbox"/> <input type="checkbox"/>
<input type="checkbox"/>	1Inf	1ero C			<input type="checkbox"/> <input type="checkbox"/>
<input type="checkbox"/>	8	8vo			<input type="checkbox"/> <input type="checkbox"/>
<input type="checkbox"/>	9	9no			<input type="checkbox"/> <input type="checkbox"/>

Mensaje de Confirmación

¿Está seguro que desea continuar?

Si No

- Materias Cursos – Permite al usuario asignar materias a cada curso de la unidad educativa mostrando la siguiente pantalla:



4.3 Materias

Para ingresar a la opción de mantenimiento de materias se debe de dar click en la opción del menú con el mismo nombre dentro del módulo de mantenimiento y se abrirá la siguiente pantalla:

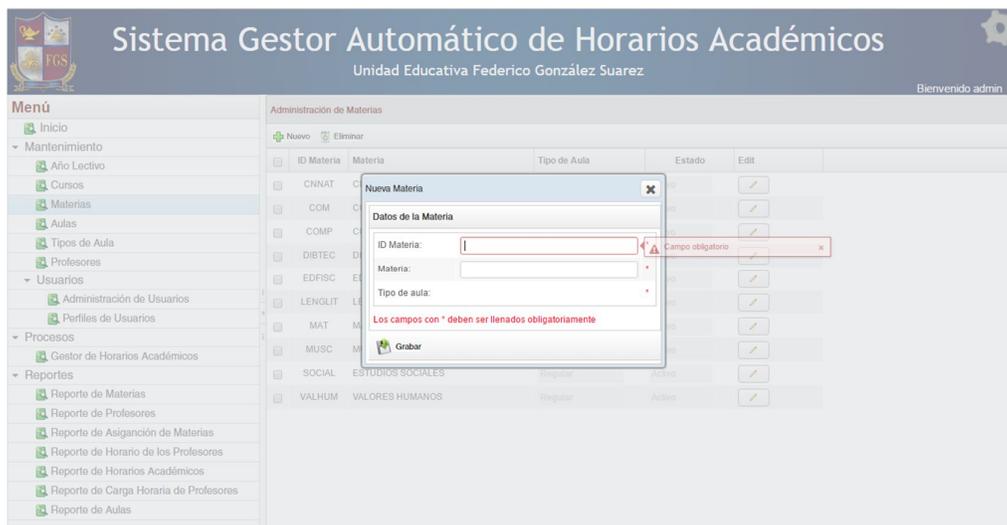




Dentro del mantenimiento de materias se encuentran los siguientes botones:



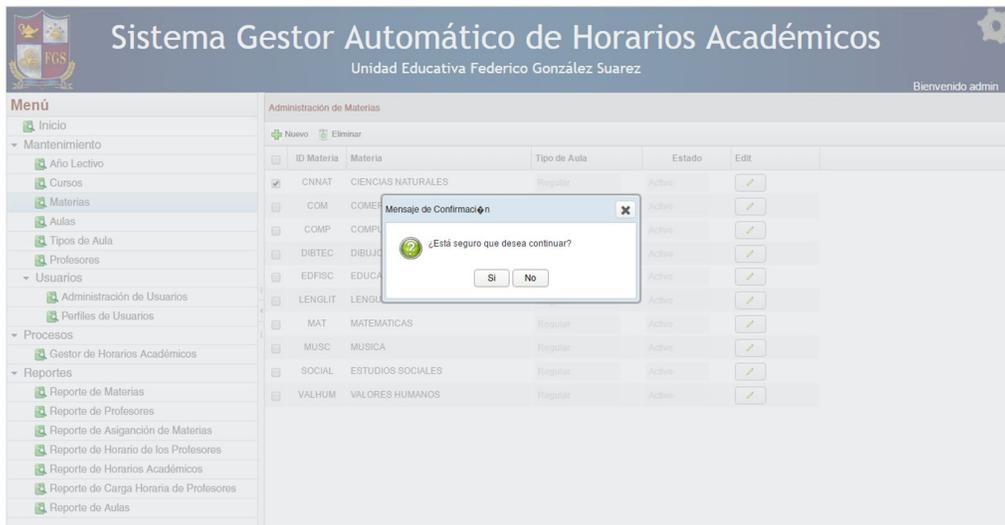
- Nuevo – Permite al usuario crear una nueva materia desplegando la siguiente pantalla:



- Editar – Permite al usuario editar la materia creada presionando en el botón  habilitando los campos de edición de la materia y mostrando los botones de  guardar o  cancelar los cambios mostrando la pantalla de la siguiente forma:



- Eliminar – Permite al usuario eliminar una materia seleccionada de la lista mostrando el siguiente mensaje de confirmación:



4.4 Aulas

Para ingresar a la opción de mantenimiento de aulas se debe de dar click en la opción del menú con el mismo nombre dentro del módulo de mantenimiento y se abrirá la siguiente pantalla:

Menú	
	Inicio
▼	Mantenimiento
	Año Lectivo
	Cursos
	Materias
	Aulas
	Tipos de Aula
	Profesores
▼	Usuarios
	Administración de Usuarios
	Perfiles de Usuarios

Sistema Gestor Automático de Horarios Académicos

Unidad Educativa Federico González Suárez

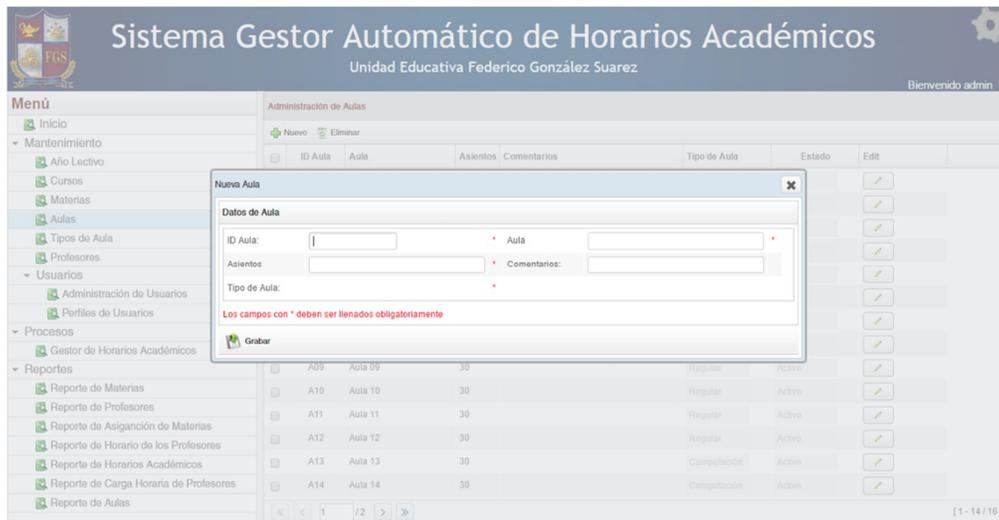
Bienvenido admin

Menú	Administración de Aulas																																																																																																									
<ul style="list-style-type: none"> Inicio ▼ Mantenimiento <ul style="list-style-type: none"> Año Lectivo Cursos Materias Aulas Tipos de Aula Profesores ▼ Usuarios <ul style="list-style-type: none"> Administración de Usuarios Perfiles de Usuarios ▼ Procesos <ul style="list-style-type: none"> Gestor de Horarios Académicos ▼ Reportes <ul style="list-style-type: none"> Reporte de Materias Reporte de Profesores Reporte de Asignación de Materias Reporte de Horario de los Profesores Reporte de Horarios Académicos Reporte de Carga Horaria de Profesores Reporte de Aulas 	<div style="border: 1px solid #ccc; padding: 5px;"> <div style="display: flex; justify-content: space-between; border-bottom: 1px solid #ccc; margin-bottom: 5px;"> </div> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th>ID Aula</th> <th>Aula</th> <th>Asientos</th> <th>Comentarios</th> <th>Tipo de Aula</th> <th>Estado</th> <th>Edit</th> </tr> </thead> <tbody> <tr><td>A01</td><td>Aula 01</td><td>30</td><td></td><td>Regular</td><td>Activo</td><td></td></tr> <tr><td>A02</td><td>Aula 02</td><td>30</td><td></td><td>Regular</td><td>Activo</td><td></td></tr> <tr><td>A03</td><td>Aula 03</td><td>30</td><td></td><td>Regular</td><td>Activo</td><td></td></tr> <tr><td>A04</td><td>Aula 04</td><td>30</td><td></td><td>Regular</td><td>Activo</td><td></td></tr> <tr><td>A05</td><td>Aula 05</td><td>30</td><td></td><td>Regular</td><td>Activo</td><td></td></tr> <tr><td>A06</td><td>Aula 06</td><td>30</td><td></td><td>Regular</td><td>Activo</td><td></td></tr> <tr><td>A07</td><td>Aula 07</td><td>30</td><td></td><td>Regular</td><td>Activo</td><td></td></tr> <tr><td>A08</td><td>Aula 08</td><td>30</td><td></td><td>Regular</td><td>Activo</td><td></td></tr> <tr><td>A09</td><td>Aula 09</td><td>30</td><td></td><td>Regular</td><td>Activo</td><td></td></tr> <tr><td>A10</td><td>Aula 10</td><td>30</td><td></td><td>Regular</td><td>Activo</td><td></td></tr> <tr><td>A11</td><td>Aula 11</td><td>30</td><td></td><td>Regular</td><td>Activo</td><td></td></tr> <tr><td>A12</td><td>Aula 12</td><td>30</td><td></td><td>Regular</td><td>Activo</td><td></td></tr> <tr><td>A13</td><td>Aula 13</td><td>30</td><td></td><td>Computación</td><td>Activo</td><td></td></tr> <tr><td>A14</td><td>Aula 14</td><td>30</td><td></td><td>Computación</td><td>Activo</td><td></td></tr> </tbody> </table> <div style="display: flex; justify-content: space-between; border-top: 1px solid #ccc; margin-top: 5px;"> << 1 / 2 >> [1 - 14 / 16] </div> </div>	ID Aula	Aula	Asientos	Comentarios	Tipo de Aula	Estado	Edit	A01	Aula 01	30		Regular	Activo		A02	Aula 02	30		Regular	Activo		A03	Aula 03	30		Regular	Activo		A04	Aula 04	30		Regular	Activo		A05	Aula 05	30		Regular	Activo		A06	Aula 06	30		Regular	Activo		A07	Aula 07	30		Regular	Activo		A08	Aula 08	30		Regular	Activo		A09	Aula 09	30		Regular	Activo		A10	Aula 10	30		Regular	Activo		A11	Aula 11	30		Regular	Activo		A12	Aula 12	30		Regular	Activo		A13	Aula 13	30		Computación	Activo		A14	Aula 14	30		Computación	Activo	
ID Aula	Aula	Asientos	Comentarios	Tipo de Aula	Estado	Edit																																																																																																				
A01	Aula 01	30		Regular	Activo																																																																																																					
A02	Aula 02	30		Regular	Activo																																																																																																					
A03	Aula 03	30		Regular	Activo																																																																																																					
A04	Aula 04	30		Regular	Activo																																																																																																					
A05	Aula 05	30		Regular	Activo																																																																																																					
A06	Aula 06	30		Regular	Activo																																																																																																					
A07	Aula 07	30		Regular	Activo																																																																																																					
A08	Aula 08	30		Regular	Activo																																																																																																					
A09	Aula 09	30		Regular	Activo																																																																																																					
A10	Aula 10	30		Regular	Activo																																																																																																					
A11	Aula 11	30		Regular	Activo																																																																																																					
A12	Aula 12	30		Regular	Activo																																																																																																					
A13	Aula 13	30		Computación	Activo																																																																																																					
A14	Aula 14	30		Computación	Activo																																																																																																					

Dentro del mantenimiento de aulas se encuentran los siguientes botones:



- Nuevo – Permite al usuario crear una nueva aula desplegando la siguiente pantalla:



- Editar – Permite al usuario editar el aula creada presionando en el botón  habilitando los campos de edición del año lectivo y mostrando los botones de  guardar o  cancelar los cambios mostrando la pantalla de la siguiente forma:



- Eliminar – Permite al usuario eliminar un aula seleccionada de la lista mostrando el siguiente mensaje de confirmación:

Sistema Gestor Automático de Horarios Académicos
Unidad Educativa Federico González Suarez

Bienvenido admin

Menú

- Inicio
- ▼ Mantenimiento
 - Año Lectivo
 - Cursos
 - Materias
 - Aulas
 - Tipos de Aula
 - Profesores
 - ▼ Usuarios
 - Administración de Usuarios
 - Perfiles de Usuarios
- ▼ Procesos
 - Gestor de Horarios Académicos
- ▼ Reportes
 - Reporte de Materias
 - Reporte de Profesoras
 - Reporte de Asignación de Materias
 - Reporte de Horario de los Profesores
 - Reporte de Horarios Académicos
 - Reporte de Carga Horaria de Profesores
 - Reporte de Aulas

Administración de Aulas

Nuevo Eliminar

ID Aula	Aula	Asientos	Comentarios	Tipo de Aula	Estado	Edit
<input checked="" type="checkbox"/>	A01	Aula 01	30	Regular	Activo	
<input type="checkbox"/>	A02	Aula 02		Regular	Activo	
<input type="checkbox"/>	A03	Aula 03		Regular	Activo	
<input type="checkbox"/>	A04	Aula 04		Regular	Activo	
<input type="checkbox"/>	A05	Aula 05		Regular	Activo	
<input type="checkbox"/>	A06	Aula 06		Regular	Activo	
<input type="checkbox"/>	A07	Aula 07	30	Regular	Activo	
<input type="checkbox"/>	A08	Aula 08	30	Regular	Activo	
<input type="checkbox"/>	A09	Aula 09	30	Regular	Activo	
<input type="checkbox"/>	A10	Aula 10	30	Regular	Activo	
<input type="checkbox"/>	A11	Aula 11	30	Regular	Activo	
<input type="checkbox"/>	A12	Aula 12	30	Regular	Activo	
<input type="checkbox"/>	A13	Aula 13	30	Computación	Activo	
<input type="checkbox"/>	A14	Aula 14	30	Computación	Activo	

1 / 2

[1 - 14 / 16]

Mensaje de Confirmación

¿Está seguro que desea continuar?

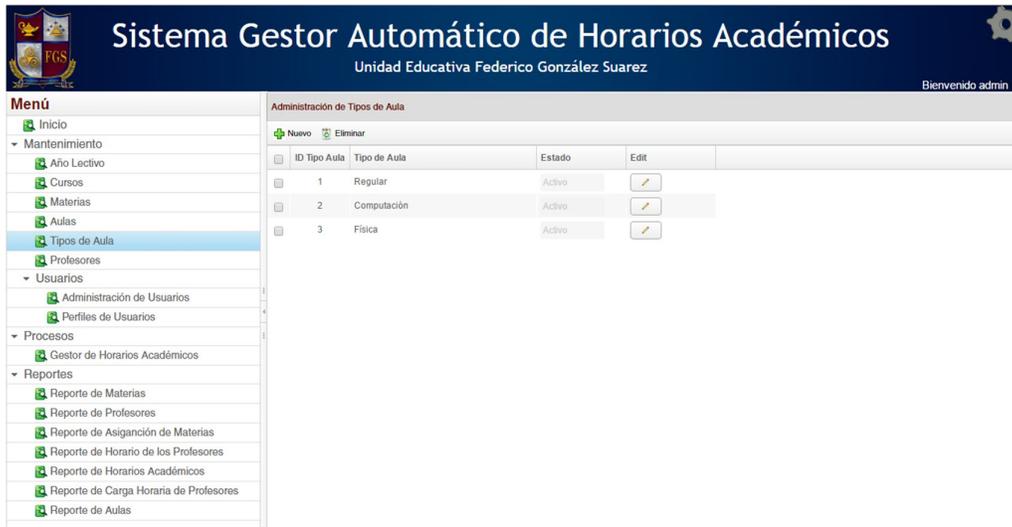
Si No

4.5 Tipos de aula

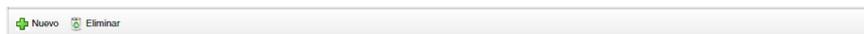
Para ingresar a la opción de mantenimiento de tipos de aula se debe de dar click en la opción del menú con el mismo nombre dentro del módulo de mantenimiento y se abrirá la siguiente pantalla:

Menú

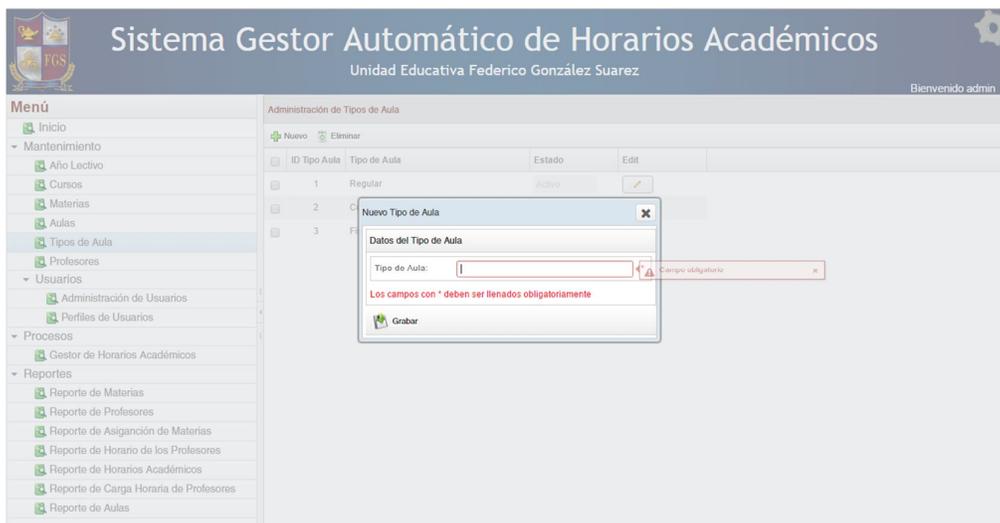
- Inicio
- ▼ Mantenimiento
 - Año Lectivo
 - Cursos
 - Materias
 - Aulas
 - Tipos de Aula
 - Profesores
 - ▼ Usuarios
 - Administración de Usuarios
 - Perfiles de Usuarios



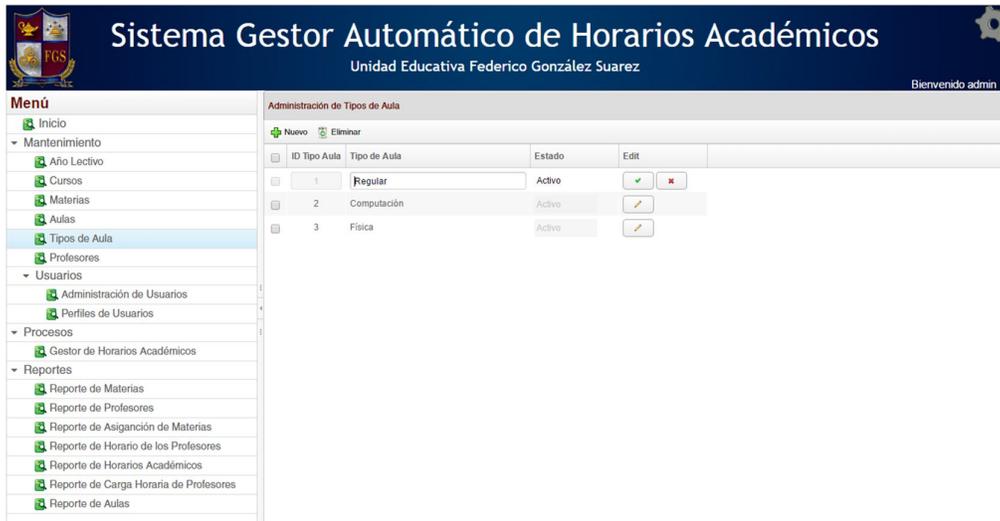
Dentro del mantenimiento del tipo de aula se encuentran los siguientes botones:



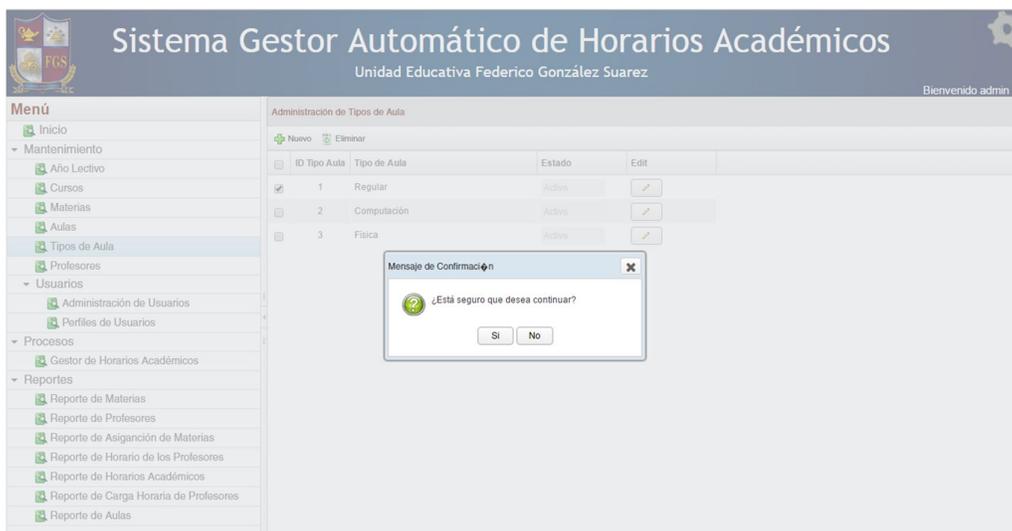
- Nuevo – Permite al usuario crear un nuevo tipo de aula desplegando la siguiente pantalla:



- Editar – Permite al usuario editar el tipo de aula creado presionando en el botón  habilitando los campos de edición del tipo de aula y mostrando los botones de  guardar o  cancelar los cambios mostrando la pantalla de la siguiente forma:



- Eliminar – Permite al usuario eliminar un tipo de aula seleccionado de la lista mostrando el siguiente mensaje de confirmación:



4.6 Profesores

Para ingresar a la opción de mantenimiento de profesores se debe de dar click en la opción del menú con el mismo nombre dentro del módulo de mantenimiento y se abrirá la siguiente pantalla:



Sistema Gestor Automático de Horarios Académicos
 Unidad Educativa Federico González Suárez

Bienvenido admin

Menú

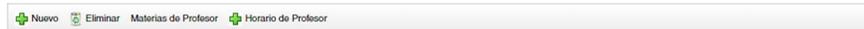
- Inicio
- Mantenimiento
 - Año Lectivo
 - Cursos
 - Materias
 - Aulas
 - Tipos de Aula
 - Profesores
 - Usuarios
 - Administración de Usuarios
 - Perfiles de Usuarios
- Procesos
 - Gestor de Horarios Académicos
- Reportes
 - Reporte de Materias
 - Reporte de Profesores
 - Reporte de Asignación de Materias
 - Reporte de Horario de los Profesores
 - Reporte de Horarios Académicos
 - Reporte de Carga Horaria de Profesores
 - Reporte de Aulas

Administración de Profesores

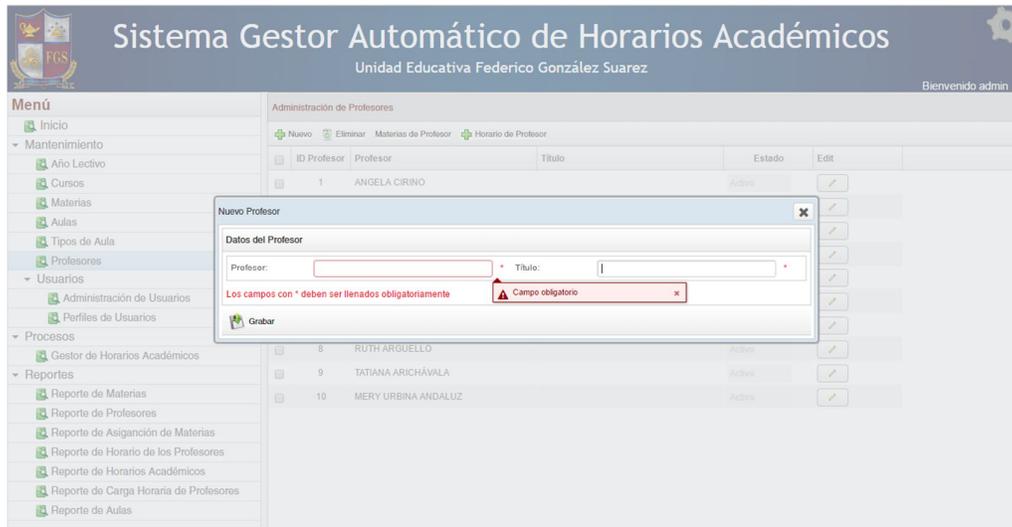
Nuevo Eliminar Materias de Profesor Horario de Profesor

ID Profesor	Profesor	Título	Estado	Edit
1	ANGELA CIRINO		Activo	
2	CAROLINA QUIROZ		Activo	
3	GLORIA ESPAÑA		Activo	
4	LUIS CAJO		Activo	
5	LUIS CORREA		Activo	
6	MARIANA CAMACHO		Activo	
7	RAUL HERRERA		Activo	
8	RUTH ARGUELLO		Activo	
9	TATIANA ARICHÁVALA		Activo	
10	MERY URBINA ANDALUZ		Activo	

Dentro del mantenimiento del profesor se encuentran los siguientes botones:



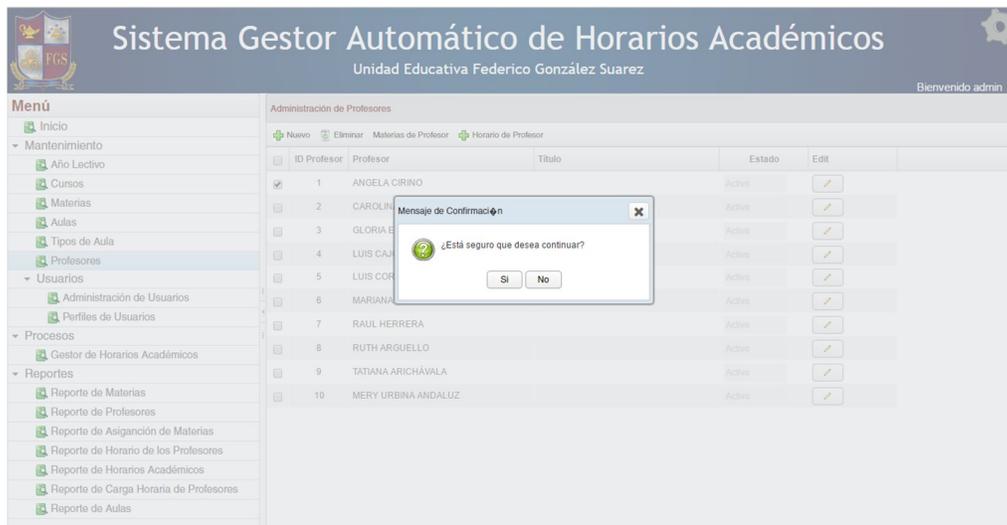
- Nuevo – Permite al usuario crear un nuevo profesores desplegando la siguiente pantalla:



- **Editar** – Permite al usuario editar el profesor creado presionando en el botón  habilitando los campos de edición del profesor y mostrando los botones de  guardar o  cancelar los cambios mostrando la pantalla de la siguiente forma:



- **Eliminar** – Permite al usuario eliminar un profesor seleccionado de la lista mostrando el siguiente mensaje de confirmación:



- Materias de Profesor – Permite al usuarios asignar las materias que dictará cada profesor mostrando la siguiente pantalla:



- Horario de Profesor – Permite al usuario configurar la disponibilidad de tiempo de cada profesor desplegando la siguiente pantalla:



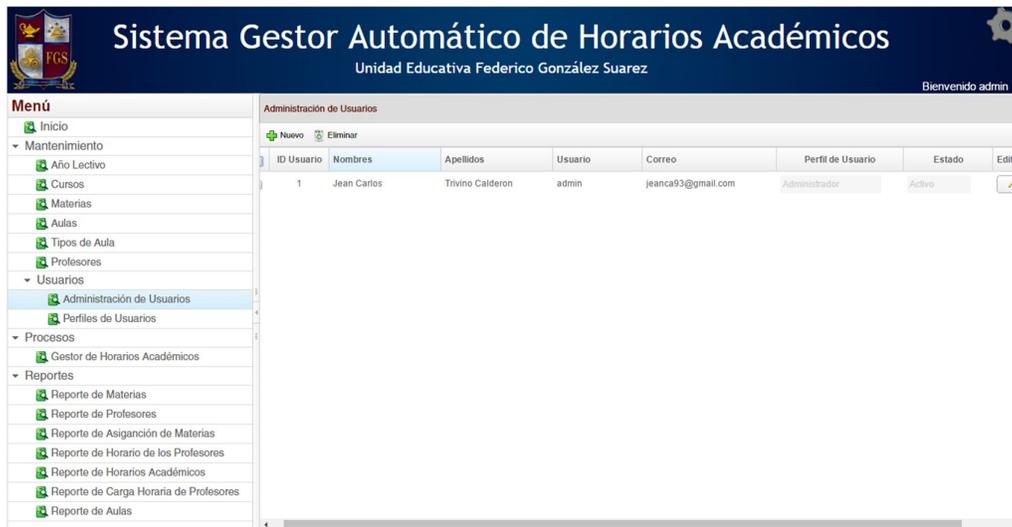
4.7 Usuarios

El sistema posee un módulo dedicado al mantenimiento y administración de usuarios para el uso y acceso al mismo por lo cual, dentro del módulo de usuarios se muestran las siguientes dos opciones:

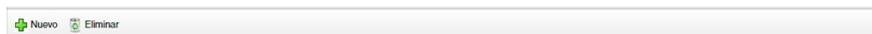
4.7.1 Administración de Usuarios

Para ingresar a la opción de administración de usuarios se debe de dar click en la opción del menú con el mismo nombre dentro del módulo de mantenimiento y se abrirá la siguiente pantalla:

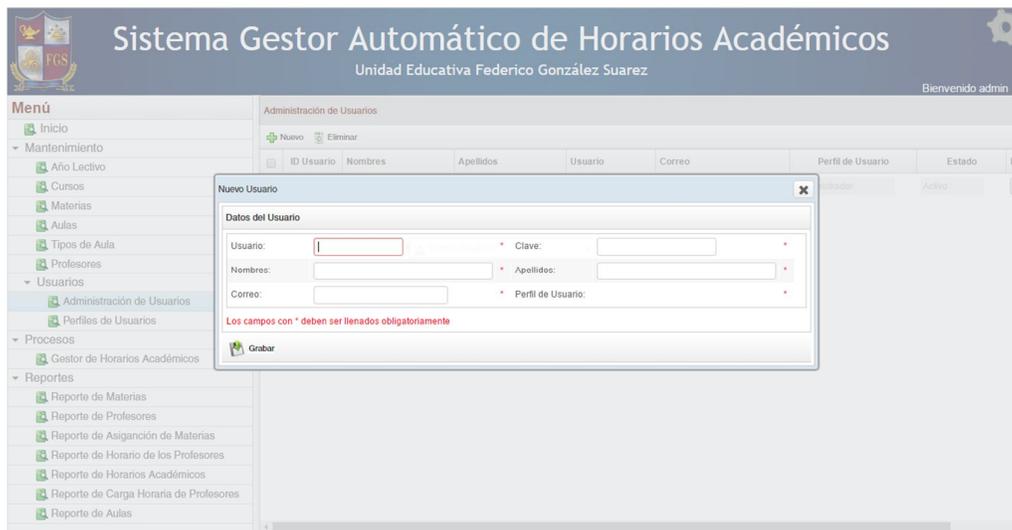




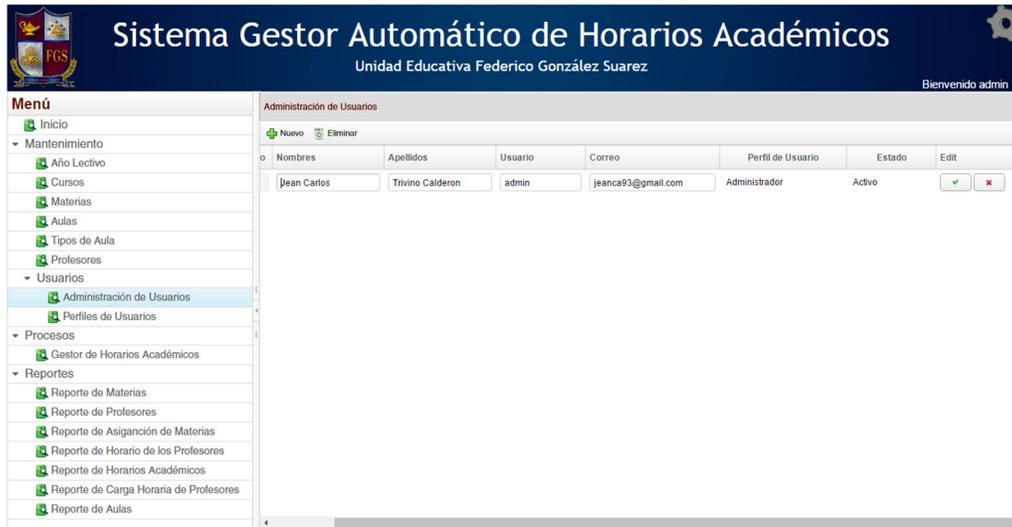
Dentro de la administración de usuarios se encuentran los siguientes botones:



- Nuevo – Permite al usuario crear un nuevo usuario desplegando la siguiente pantalla:



- Editar – Permite al usuario editar el usuario creado presionando en el botón  habilitando los campos de edición del usuario y mostrando los botones de  guardar o  cancelar los cambios mostrando la pantalla de la siguiente forma:

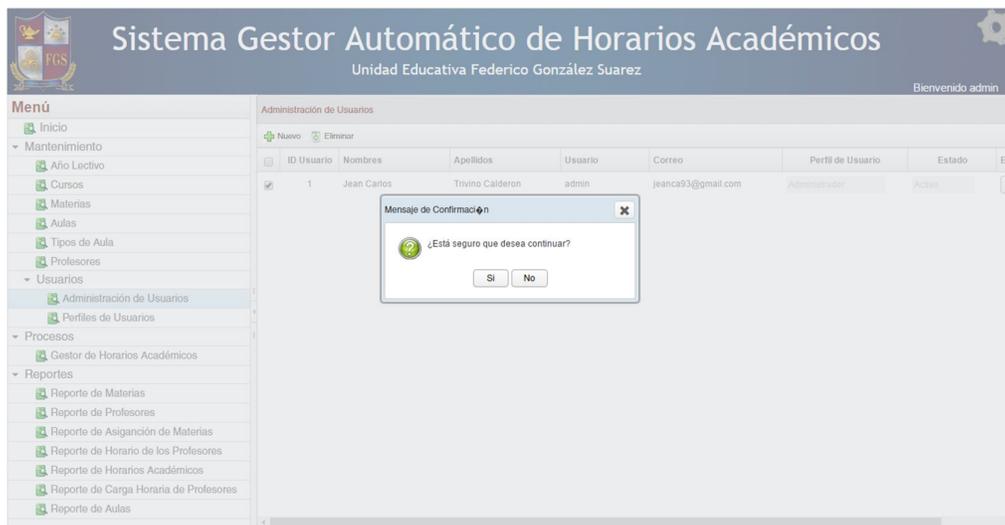


- Cambio de contraseña – Dentro de la pantalla principal del sistema se encuentra en la parte superior derecha el ícono  que le brinda la posibilidad al usuario de poder cambiar la contraseña del usuario con el que se encuentra activo en ese momento en el sistema o cerrar la sesión:



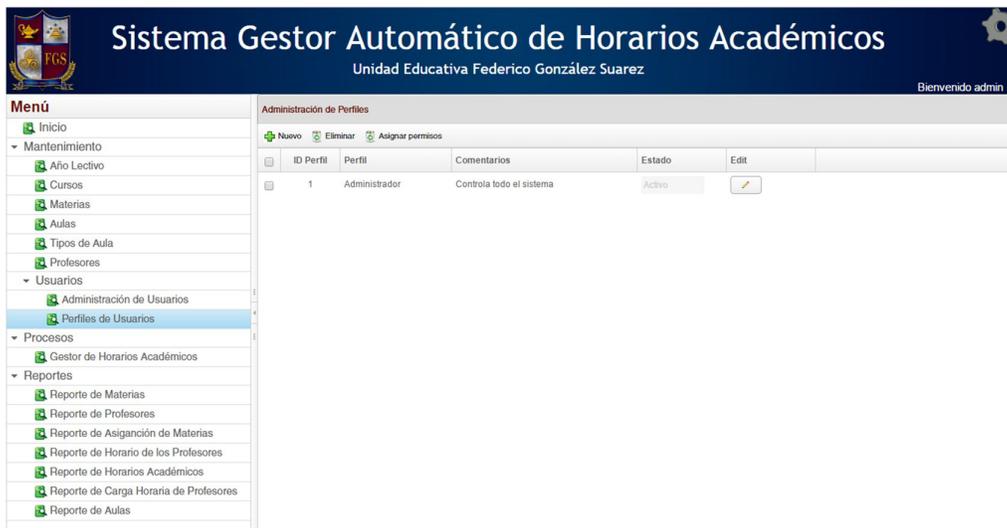


- Eliminar – Permite al usuario eliminar un usuario seleccionado de la lista mostrando el siguiente mensaje de confirmación:



4.7.2 Perfiles de Usuarios

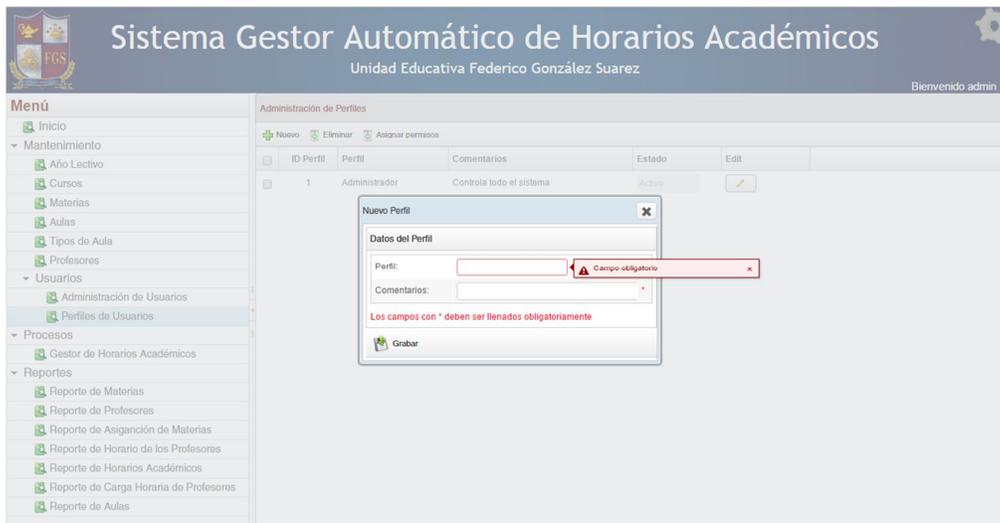
Para ingresar a la opción de mantenimiento de perfiles de usuarios se debe de dar click en la opción del menú con el mismo nombre dentro del módulo de mantenimiento y se abrirá la siguiente pantalla:



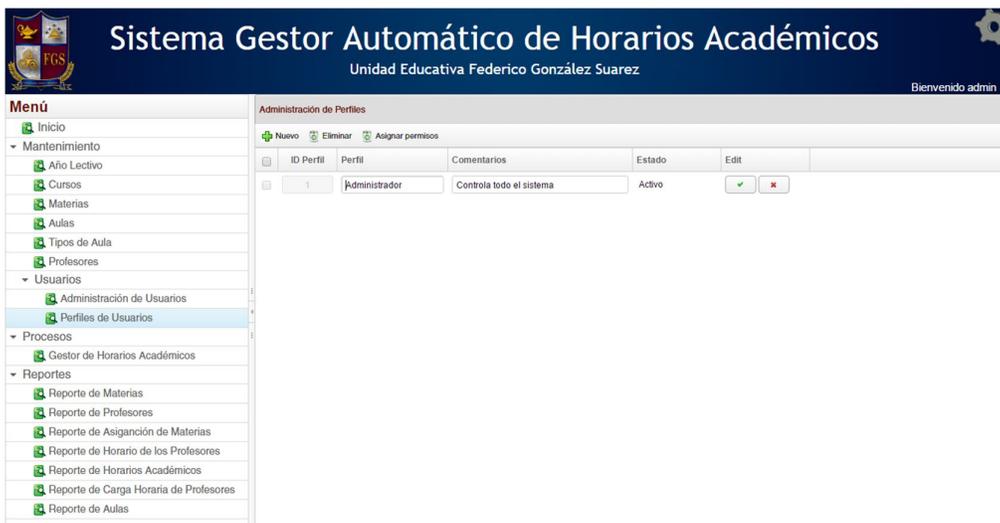
Dentro del mantenimiento de perfiles de usuarios se encuentran los siguientes botones:



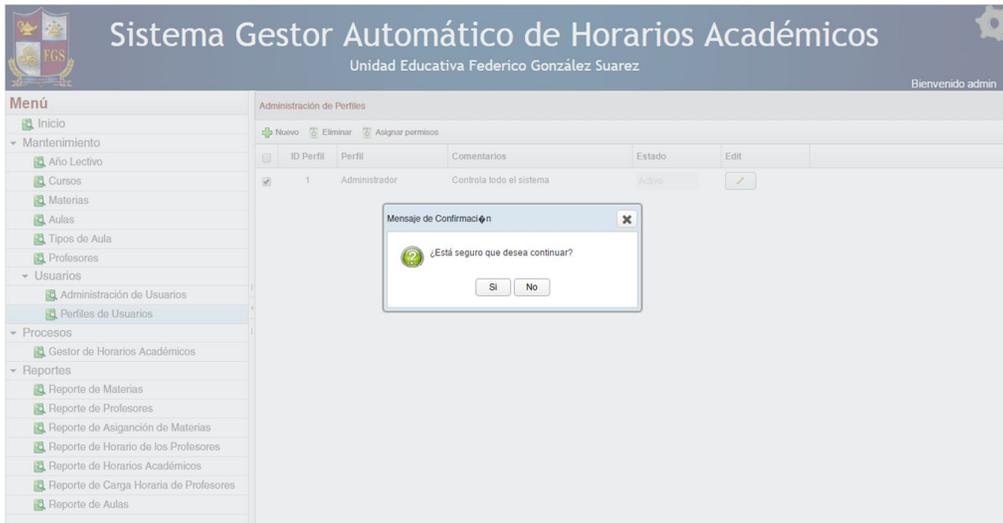
- Nuevo – Permite al usuario crear un nuevo perfil de usuario desplegando la siguiente pantalla:



- Editar – Permite al usuario editar el perfil de usuario creado presionando en el botón  habilitando los campos de edición del perfil de usuario y mostrando los botones de  guardar o  cancelar los cambios mostrando la pantalla de la siguiente forma:



- Eliminar – Permite al usuario eliminar un perfil de usuario seleccionado de la lista mostrando el siguiente mensaje de confirmación:



- Asignar permisos – Permite al usuario asignar a qué opción podrá tener acceso un usuario dentro del sistema desplegando la siguiente pantalla:

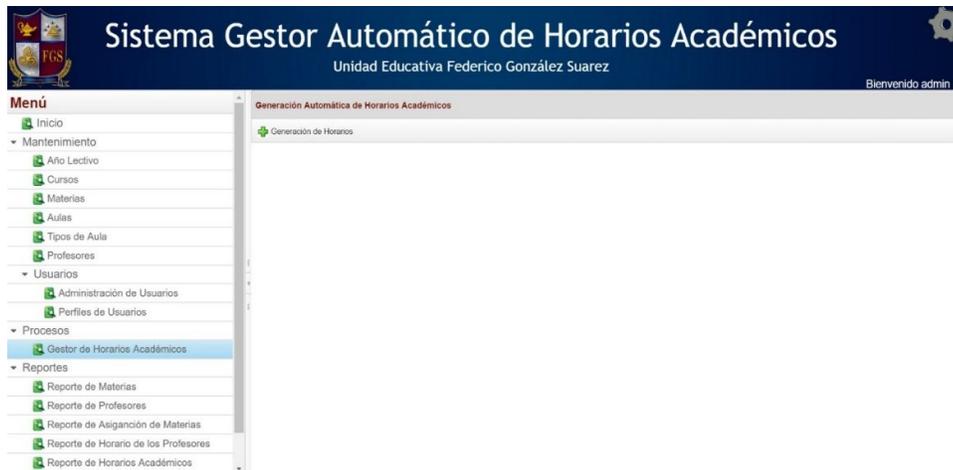


5. Proceso

5.1 Gestor de Horarios Académicos

Para ingresar a la opción de Gestor de Horarios Académicos se debe de dar click en la opción del menú con el mismo nombre dentro del módulo de Procesos y se abrirá la siguiente pantalla:

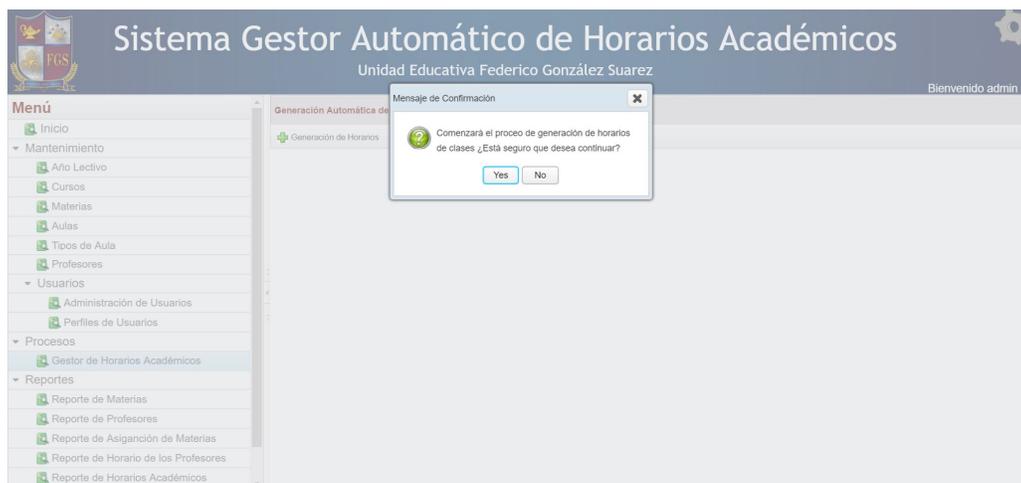




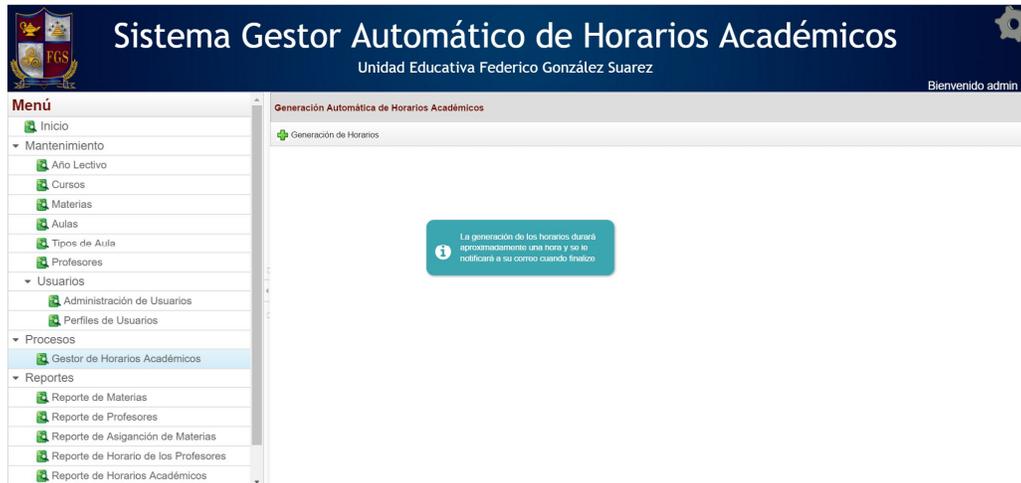
Dentro del proceso de la generación de horarios académicos se encuentra el siguiente botón:



- Generación de Horarios – Al dar clic en la opción se mostrará un mensaje de confirmación al usuario para asegurar que desea continuar con el proceso.



Una vez aceptado empezará la generación automática de los horarios académicos en segundo plano, mediante un proceso que notificará al usuario cuando este termine enviándole un mensaje al correo electrónico registrado, una vez que este concluya el botón del proceso volverá a estar visible para su ejecución.



6. Reportes

En este menú se encuentran los diferentes reportes del sistema para el cual todos tienen la opción de Exportar a Excel y PDF. Se debe de dar click en la opción del reporte deseado de la lista mencionada a continuación:

- Reporte de Materias
- Reporte de Profesores
- Reporte de Asignación de materias
- Reporte de Horarios de Profesores
- Reporte de Horarios Académicos
- Reporte de Carga Horaria de Profesores
- Reporte de Aulas

Materia	Horas Semana	Tipo de Aula	Estado
Curso 8vo			
EDUCACION FISICA	6	Regular	Activo
MATEMATICAS	2	Regular	Activo
ESTUDIOS SOCIALES	5	Regular	Activo
COMERCIO	2	Regular	Activo
DIBUJO TENICO	1	Fisica	Activo
LENGUA Y LITERATURA	2	Regular	Activo
MUSICA	6	Regular	Activo
CIENCIAS NATURALES	2	Regular	Activo

Sistema Gestor Automático de Horarios Académicos
 Unidad Educativa Federico González Suarez

Bienvenido admin

Menú

- Inicio
- Mantenimiento
 - Año Lectivo
 - Cursos
 - Materias
 - Aulas
 - Tipos de Aula
 - Profesores
- Usuarios
 - Administración de Usuarios
 - Perfiles de Usuarios
- Procesos
 - Gestor de Horarios Académicos
- Reportes
 - Reporte de Materias
 - Reporte de Profesores
 - Reporte de Asignación de Materias
 - Reporte de Horario de los Profesores

Reporte de Materias

Exportar a Excel

UNIDAD EDUCATIVA FEDERICO GONZALEZ SUAREZ

Reportes de Materias

Materia	Horas Semana	Tipo de Aula	Estado
Curso: 8vo			
EDUCACION FISICA	6	Regular	Activo
MATEMATICAS	2	Regular	Activo
ESTUDIOS SOCIALES	5	Regular	Activo
COMERCIO	2	Regular	Activo
DIBUJO TECNICO	1	Física	Activo

ReporteMaterias.xls

Sistema Gestor Automático de Horarios Académicos
 Unidad Educativa Federico González Suarez

Bienvenido admin

Menú

- Inicio
- Mantenimiento
 - Año Lectivo
 - Cursos
 - Materias
 - Aulas
 - Tipos de Aula
 - Profesores
- Usuarios
 - Administración de Usuarios
 - Perfiles de Usuarios
- Procesos
 - Gestor de Horarios Académicos
- Reportes
 - Reporte de Materias
 - Reporte de Profesores
 - Reporte de Asignación de Materias
 - Reporte de Horario de los Profesores
 - Reporte de Horarios Académicos

Reporte de Profesores

Exportar a Excel

UNIDAD EDUCATIVA FEDERICO GONZALEZ SUAREZ

Reportes de Profesores

#	Nombres	Titulo	Estado
1	ANGELA CIRINO		Activo
2	CAROLINA QUIROZ		Activo
3	GLORIA ESPAÑA		Activo
4	LUIS CAJO		Activo
5	LUIS CORREA		Activo
6	MARIANA CAMACHO		Activo
7	RAUL HERRERA		Activo
8	RUTH ARGUELLO		Activo
9	TATIANA ARICHÁVALA		Activo
10	MERY URBINA ANDALUZ		Activo

Sistema Gestor Automático de Horarios Académicos
 Unidad Educativa Federico González Suarez

Bienvenido admin

Menú

- Inicio
- Mantenimiento
 - Año Lectivo
 - Cursos
 - Materias
 - Aulas
 - Tipos de Aula
 - Profesores
- Usuarios
 - Administración de Usuarios
 - Perfiles de Usuarios
- Procesos
 - Gestor de Horarios Académicos
- Reportes
 - Reporte de Materias
 - Reporte de Profesores
 - Reporte de Asignación de Materias
 - Reporte de Horario de los Profesores

Reporte de Materias

Exportar a Excel

UNIDAD EDUCATIVA FEDERICO GONZALEZ SUAREZ

Reportes de Materias

Materia	Horas Semana	Tipo de Aula	Estado
Curso 8vo			
VALORES HUMANOS	5	Regular	Activo
COMPUTACION	1	Computación	Activo
EDUCACION FISICA	6	Regular	Activo
MATEMATICAS	2	Regular	Activo
ESTUDIOS SOCIALES	5	Regular	Activo
COMERCIO	2	Regular	Activo
DIBUJO TENICO	1	Física	Activo

ReporteMaterias.pdf

Show all downloads...

Sistema Gestor Automático de Horarios Académicos
 Unidad Educativa Federico González Suarez

Bienvenido admin

Menú

- Inicio
- Mantenimiento
 - Año Lectivo
 - Cursos
 - Materias
 - Aulas
 - Tipos de Aula
 - Profesores
- Usuarios
 - Administración de Usuarios
 - Perfiles de Usuarios
- Procesos
 - Gestor de Horarios Académicos
- Reportes
 - Reporte de Materias
 - Reporte de Profesores
 - Reporte de Asignación de Materias
 - Reporte de Horario de los Profesores

Reporte de Profesores

Exportar a Excel

UNIDAD EDUCATIVA FEDERICO GONZALEZ SUAREZ

Reportes de Profesores

#	Nombres	Título	Estado
1	ANGELA CIRINO		Activo
2	CAROLINA QUIROZ		Activo
3	GLORIA ESPAÑA		Activo
4	LUIS CAJO		Activo
5	LUIS CORREA		Activo
6	MARIANA CAMACHO		Activo
7	RAUL HERRERA		Activo
8	RUTH ARGUELLO		Activo
9	TATIANA ARICHÁVAI A		Activo

ReporteProfesores.xls

Show all downloads...

Sistema Gestor Automático de Horarios Académicos
 Unidad Educativa Federico González Suarez

Bienvenido admin

Menú

- Inicio
- Mantenimiento
 - Año Lectivo
 - Cursos
 - Materias
 - Aulas
 - Tipos de Aula
 - Profesores
- Usuarios
 - Administración de Usuarios
 - Perfiles de Usuarios
- Procesos
 - Gestor de Horarios Académicos
- Reportes
 - Reporte de Materias
 - Reporte de Profesores
 - Reporte de Asignación de Materias
 - Reporte de Horario de los Profesores

Reporte de Profesores

Exportar a Excel

UNIDAD EDUCATIVA FEDERICO GONZALEZ SUAREZ

Reportes de Profesores

#	Nombres	Titulo	Estado
1	ANGELA CIRINO		Activo
2	CAROLINA QUIROZ		Activo
3	GLORIA ESPAÑA		Activo
4	LUIS CAJO		Activo
5	LUIS CORREA		Activo
6	MARIANA CAMACHO		Activo
7	RAUL HERRERA		Activo
8	RUTH ARGUELLO		Activo
9	TATIANA ARICHÁVAI A		Activo

Show all downloads...

Sistema Gestor Automático de Horarios Académicos
 Unidad Educativa Federico González Suarez

Bienvenido admin

Menú

- Inicio
- Mantenimiento
 - Año Lectivo
 - Cursos
 - Materias
 - Aulas
 - Tipos de Aula
 - Profesores
- Usuarios
 - Administración de Usuarios
 - Perfiles de Usuarios
- Procesos
 - Gestor de Horarios Académicos
- Reportes
 - Reporte de Materias
 - Reporte de Profesores
 - Reporte de Asignación de Materias
 - Reporte de Horario de los Profesores
 - Reporte de Horarios Académicos

Reporte de Asignación de Materias

Exportar a Excel

UNIDAD EDUCATIVA FEDERICO GONZALEZ SUAREZ

Reportes de Asignación de Materias

Materias	Horas Semana	Curso	Paralelos
Profesor ANGELA CIRINO			
COMPUTACION	1	9no	2
COMPUTACION	1	8vo	2
Profesor CAROLINA QUIROZ			
DIBUJO TENICO	1	8vo	2
DIBUJO TENICO	1	9no	2
Profesor GLORIA ESPAÑA			
VALORES HUMANOS	5	9no	2
VALORES HUMANOS	5	8vo	2

Sistema Gestor Automático de Horarios Académicos
Unidad Educativa Federico González Suárez

Bienvenido admin

Menú

- Inicio
- Mantenimiento
 - Año Lectivo
 - Cursos
 - Materias
 - Aulas
 - Tipos de Aula
 - Profesores
- Usuarios
 - Administración de Usuarios
 - Perfiles de Usuarios
- Procesos
 - Gestor de Horarios Académicos
- Reportes
 - Reporte de Materias
 - Reporte de Profesores
 - Reporte de Asignación de Materias
 - Reporte de Horario de los Profesores

Reporte de Asignación de Materias

Exportar a Excel

UNIDAD EDUCATIVA FEDERICO GONZALEZ SUAREZ

Reportes de Asignación de Materias

Materias	Horas Semana	Curso	Paralelos
Profesor: ANGELA CIRINO			
COMPUTACION	1	9no	2
COMPUTACION	1	8vo	2
Profesor: CAROLINA QUIROZ			
DIBUJO TENICO	1	8vo	2
DIBUJO TENICO	1	9no	2
Profesor: GLORIA ESPAÑA			
VALORES HUMANOS	5	9no	2

Show all downloads...

Sistema Gestor Automático de Horarios Académicos
Unidad Educativa Federico González Suárez

Bienvenido admin

Menú

- Inicio
- Mantenimiento
 - Año Lectivo
 - Cursos
 - Materias
 - Aulas
 - Tipos de Aula
 - Profesores
- Usuarios
 - Administración de Usuarios
 - Perfiles de Usuarios
- Procesos
 - Gestor de Horarios Académicos
- Reportes
 - Reporte de Materias
 - Reporte de Profesores
 - Reporte de Asignación de Materias
 - Reporte de Horario de los Profesores

Reporte de Asignación de Materias

Exportar a Excel

UNIDAD EDUCATIVA FEDERICO GONZALEZ SUAREZ

Reportes de Asignación de Materias

Materias	Horas Semana	Curso	Paralelos
Profesor: ANGELA CIRINO			
COMPUTACION	1	9no	2
COMPUTACION	1	8vo	2
Profesor: CAROLINA QUIROZ			
DIBUJO TENICO	1	8vo	2
DIBUJO TENICO	1	9no	2
Profesor: GLORIA ESPAÑA			
VALORES HUMANOS	5	9no	2

Show all downloads...

Sistema Gestor Automático de Horarios Académicos
 Unidad Educativa Federico González Suarez

Bienvenido admin

Inicio

- Mantenimiento
 - Año Lectivo
 - Cursos
 - Materias
 - Aulas
 - Tipos de Aula
 - Profesores
- Usuarios
 - Administración de Usuarios
 - Perfiles de Usuarios
- Procesos
 - Gestor de Horarios Académicos
- Reportes
 - Reporte de Materias
 - Reporte de Profesores
 - Reporte de Asignación de Materias
 - Reporte de Horario de los Profesores
 - Reporte de Horarios Académicos
 - Reporte de Carga Horaria de Profesores

Reporte de Horario de Profesores

Exportar a Excel

UNIDAD EDUCATIVA FEDERICO GONZALEZ SUAREZ

Reportes de Horarios de Profesores

	Hora de Inicio	Hora Finalización
Profesor	ANGELA CIRINO	
Día	Lunes	
	07:00	07:45
	07:45	08:30
	08:30	09:15
	09:15	10:00
Día	Martes	
	08:30	09:15
Día	Miercoles	

Sistema Gestor Automático de Horarios Académicos
 Unidad Educativa Federico González Suarez

Bienvenido admin

Inicio

- Mantenimiento
 - Año Lectivo
 - Cursos
 - Materias
 - Aulas
 - Tipos de Aula
 - Profesores
- Usuarios
 - Administración de Usuarios
 - Perfiles de Usuarios
- Procesos
 - Gestor de Horarios Académicos
- Reportes
 - Reporte de Materias
 - Reporte de Profesores
 - Reporte de Asignación de Materias
 - Reporte de Horario de los Profesores
 - Reporte de Horarios Académicos

Reporte de Horario de Profesores

Exportar a Excel

UNIDAD EDUCATIVA FEDERICO GONZALEZ SUAREZ

Reportes de Horarios de Profesores

	Hora de Inicio	Hora Finalización
Profesor	ANGELA CIRINO	
Día	Lunes	
	07:00	07:45
	07:45	08:30
	08:30	09:15
	09:15	10:00
Día	Martes	
	08:30	09:15

ReporteHorarioProfes...xls

Show all downloads

Sistema Gestor Automático de Horarios Académicos
 Unidad Educativa Federico González Suárez

Bienvenido admin

Inicio

- Mantenimiento
 - Año Lectivo
 - Cursos
 - Materias
 - Aulas
 - Tipos de Aula
 - Profesores
- Usuarios
 - Administración de Usuarios
 - Perfiles de Usuarios
- Procesos
 - Gestor de Horarios Académicos
- Reportes
 - Reporte de Materias
 - Reporte de Profesores
 - Reporte de Asignación de Materias
 - Reporte de Horario de los Profesores
 - Reporte de Horarios Académicos

Reporte de Horario de Profesores

Exportar a Excel

UNIDAD EDUCATIVA FEDERICO GONZALEZ SUAREZ

Reportes de Horarios de Profesores

	Hora de Inicio	Hora Finalización
Profesor	ANGELA CIRINO	
Día	Lunes	
	07:00	07:45
	07:45	08:30
	08:30	09:15
	09:15	10:00
Día	Martes	
	08:30	09:15

ReporteHorarioProfe...pdf

Show all downloads...

6.4 Anexo 4: Manual Técnico

Contenido:

1 Arquitectura

1.1 JRE

1.2 JBOSS

1.3 Maven

1.4 ZK

1.5 Eclipse Luna

1.6 Configurando variables de entorno

2 Servicio Web

3 GAMS

3.1 Modelo Matemático

3.2 Script Modelo Matemático

4 Base de Datos

4.1 Credenciales

4.2 Script de generación

4.3 Diccionario de Datos

4.4 Modelo Entidad Relación

1. Arquitectura del desarrollo

Para el desarrollo del sistema web se utiliza como lenguaje de programación AJAX utilizando Eclipse como ambiente de desarrollo, basándose en una arquitectura de solución de Cliente/Servidor en 3 capas, utilizando como framework ZK. Adicional a esto para la gestión de la información del colegio se manejará un motor de base de datos potente y gratuito como lo es MySQL, que con ayuda de JBOSS como servidor web se manejará la conexión respectiva con el sistema utilizando Hibernate para mayor seguridad implementado HQL orientado a objetos donde realiza un mapeo de las tablas de la base de datos para ser utilizadas como clases siendo esto un lenguaje SQL más estructurado.

1.1 JRE

Se debe tener instalado la versión 7u79 de las herramientas de desarrollo y entorno de ejecución de Java necesario para el correcto funcionamiento del sistema propuesto, mismo que contiene el jdk correspondiente para el desarrollo.

1.2 JBOSS

Se debe descargar la última versión estable del servidor web de código abierto basado en Java puro utilizado para el desarrollo del sistema la cual es JBoss AS 7.1.1.Final, mismo que puede ser instalado en cualquier sistema operativo.

1.2.1 Configuración de JBOSS

En el directorio `jboss-as-7.1.1.Final\standalone\configuration` se debe añadir al archivo `standalone-full.xml` el siguiente bloque donde se especifica la fuente de la información especificada para el sistema.

```
<datasources>
  <datasource jndi-name="java:jboss/datasources/ExampleDS" pool-name="ExampleDS"
    enabled="true" use-java-context="true">
    <connection-url>jdbc:h2:mem:test;DB_CLOSE_DELAY=-1</connection-url>
    <driver>h2</driver>
    <security>
      <user-name>sa</user-name>
      <password>sa</password>
    </security>
  </datasource>
  <datasource jndi-name="java:/REPORTES" pool-name="REPORTES">
    <connection-url>jdbc:mysql://localhost:3306/gaha</connection-url>
    <driver>mysql</driver>
    <security>
      <user-name>root</user-name>
      <password>GAHA2016</password>
    </security>
  </datasource>
</drivers>
  <driver name="h2" module="com.h2database.h2">
    <xa-datasource-class>org.h2.jdbcx.JdbcDataSource</xa-datasource-class>
  </driver>
  <driver name="mysql" module="com.mysql"/>
</drivers>
</datasources>
```

1.3 Maven

Para la gestión de las distintas librerías y complementos que se utilizará en el sistema, por motivos de optimización de recursos y facilidad en la implementación se utiliza la versión Apache Maven 3.2.1 para la gestión y construcción de proyectos Java.

1.4 ZK

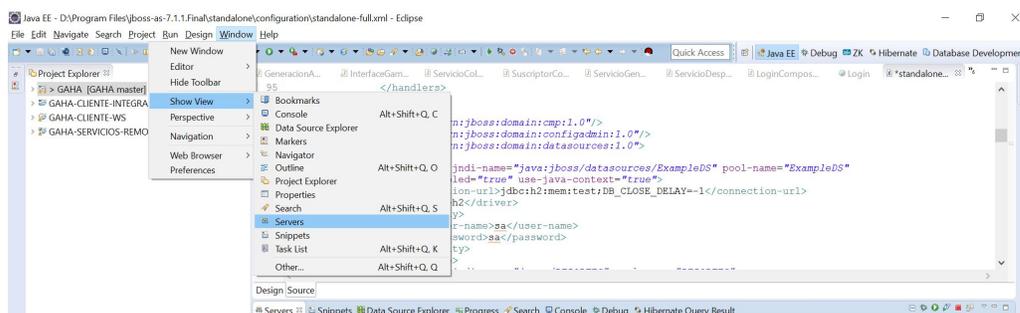
Para el desarrollo del sistema se utilizó la versión más actual y estable al momento de ZK la cual es 8.0.2.1, utilizando todos los complementos, librerías y bondades que esta proporciona para su funcionamiento

1.5 Eclipse Luna

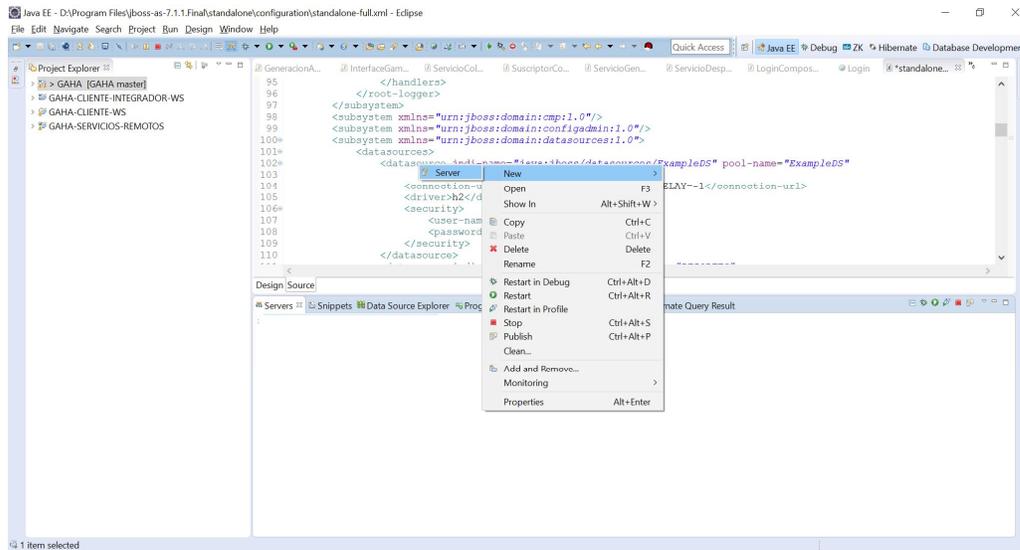
El ambiente de desarrollo utilizado para el desarrollo del sistema es eclipse luna para Windows en su versión de 32 o 64 bits dependiendo del sistema operativo a utilizar, teniendo en consideración que se debe tener la misma versión del JRE.

1.5.1 Configuración de Eclipse

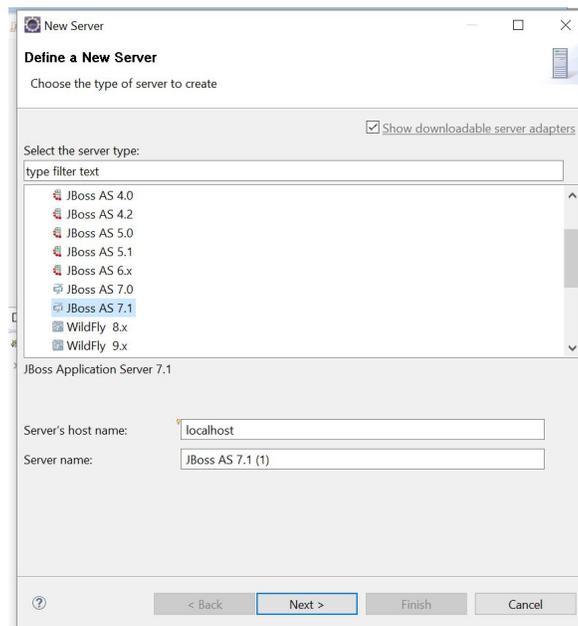
Una vez instalado eclipse, ejecutar y dirigirse al menú superior, buscar la opción Windows, luego ver vista y dar click en servidor.



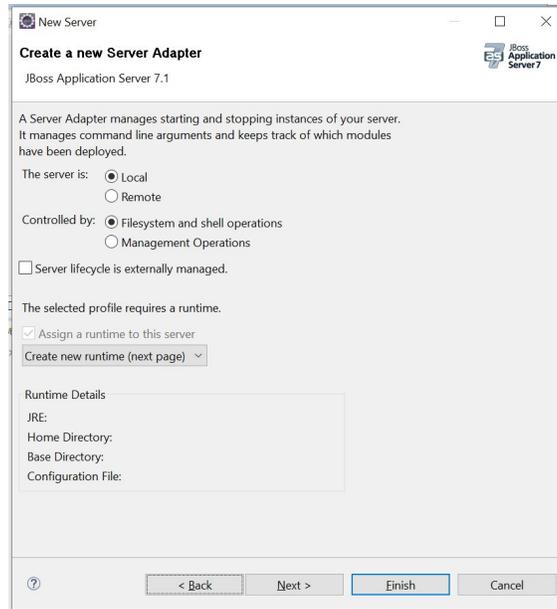
Sobre la vista de servidores dar click derecho, luego escoger la opción nuevo y por ultimo servidor.



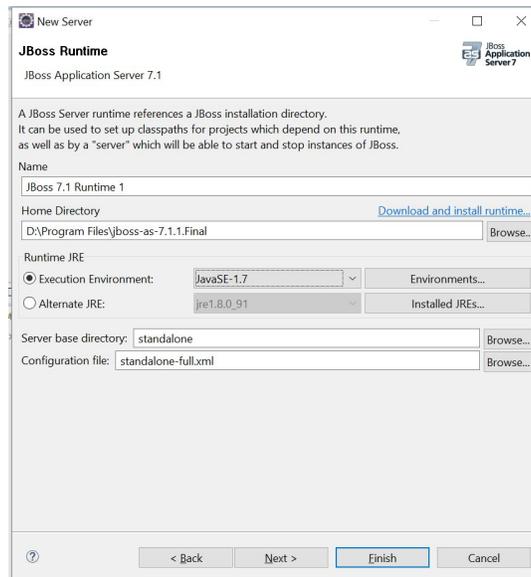
De la vista de servidores se escoge JBOSS AS 7.1 y se le da un nombre al servidor.



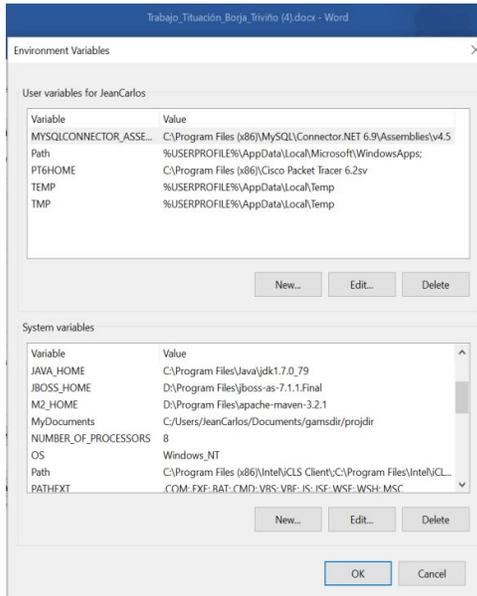
Luego se mostrará la siguiente ventana y dar click en siguiente



Se debe especificar la ruta donde se descargó el servidor JBOSS y posteriormente seleccionar la versión correspondiente al JRE descargado como se indica en la imagen.

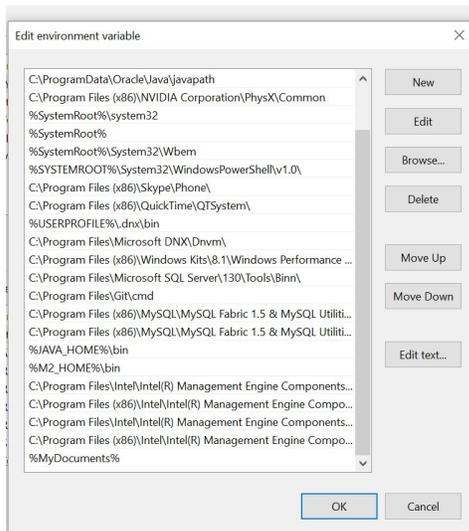


1.6 Configurando variables de entorno



1) Se debe configurar las variables de entorno correspondientes para Java, JBOSS y Maven necesarias:

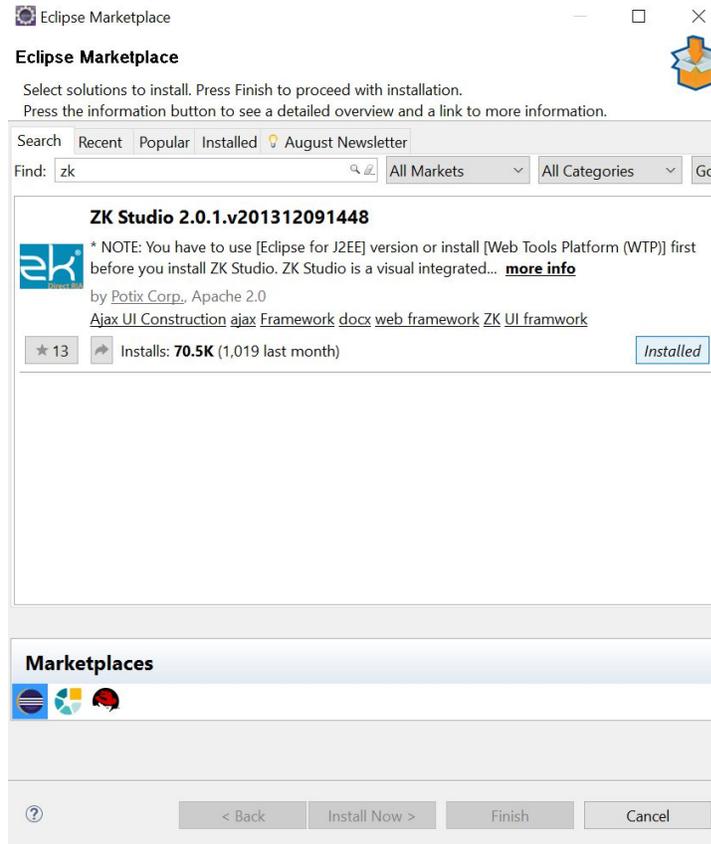
- JAVA_HOME: [Ruta de instalación del jdk]
- JBOSS_HOME: [Ruta donde se descargó la carpeta del servidor web]
- M2_HOME: [Ruta donde se descargó la carpeta del maven]
- MyDocuments: [Ruta absoluta del directorio de proyectos con el que trabaja GAMS]



2) Se debe configurar las rutas de fácil acceso en la variable de entorno path correspondientes para Java y Maven necesarias:

- Java > %JAVA_HOME%\bin
- Maven > %M2_HOME%\bin
- MyDocuments > %MyDocuments%

Al ejecutar eclipse, dirigirse a Windows > Eclipse Marketplace y descargar la herramienta necesaria para el desarrollo ZK Studio



2. Servicio Web

Se realizo la creación de un servicio web para realizar el proceso de generación de horarios el cual consiste en dos funciones utilizando el método SOAP de manera asincrónica, la primera que es la generación del archivo tipo Excel que recibe como parámetros la información en forma de tramas que envía el cliente, en este caso el proyecto del sistema y la segunda que es la llamada al proceso de consola al cual se le envían los comandos correspondientes para ejecutar GAMS de manera remota, para esto se utiliza la variable de entorno previamente configrada MyDocuments

para posicionar el directorio en donde se encuentran los documentos dentro del archivo Excel como el modelo matemático utilizado en GAMS.

3. GAMS

Para el desarrollo del sistema se utilizó la versión 23.5.2 de GAMS en su versión del sistema operativo Windows ya sea de 32 o 64 bits, utilizando todos los complementos, librerías y bondades que esta proporciona para su funcionamiento.

3.1 Modelo matemático

Se desarrolló el siguiente modelo matemático para la generación de horarios académico considerando toda la información del colegio:

- Profesores
- Materias
- Aulas
- Tipos de Aulas
- Cursos
- Paralelos
- Días
- Periodos

En base a esta información se ha tenido las siguientes consideraciones:

- Disponibilidad de los profesores fundamental para el proceso de generación debido que en base a esto se generan los horarios.
- Los paralelos que se van a aperturar por cursos.
- La malla curricular del plantel.
- La cantidad de horas a la semana que se deben ver las materias por curso.
- Las materias por curso que pueden impartir los profesores.
- La categorización de las aulas depende de su tipo o uso.
- El tipo de aula que requiere la materia para dar clase.

El modelo recibe como entrada / salida un archivo tipo Excel, el cual contendrá diferentes pestañas de donde se leerán tanto los índices como parámetros requeridos y al finalizar el proceso generara varias pestañas de resultados para poder apreciar desde diferentes perspectivas la generación del horario. El sistema utilizara para su proceso la última pestaña que se denominara Resultados X.

3.2 Script Modelo Matemático

```

set p profesores /
$call =xls2gms r=indices!a2:a20 i=nuevo.xlsx o=set1.inc b=1
$include set1.inc
/;
set m materias /
$call =xls2gms r=indices!b2:b10000 i=nuevo.xlsx o=set1.inc b=1
$include set1.inc
/;
set t tipo de aula /
$call =xls2gms r=indices!c2:c10000 i=nuevo.xlsx o=set1.inc b=1
$include set1.inc
/;

set k aula /
$call =xls2gms r=indices!d2:d10000 i=nuevo.xlsx o=set1.inc b=1
$include set1.inc
/;
set c curso /
$call =xls2gms r=indices!e2:e10000 i=nuevo.xlsx o=set1.inc b=1
$include set1.inc
/;
set l paralelo /
$call =xls2gms r=indices!f2:f10000 i=nuevo.xlsx o=set1.inc
$include set1.inc
/;
set d dias de la semana escolares/
$call =xls2gms r=indices!g2:g10000 i=nuevo.xlsx o=set1.inc b=1
$include set1.inc
/;
set h hora de clase/
$call =xls2gms r=indices!h2:h10000 i=nuevo.xlsx o=set1.inc
$include set1.inc
/;
alias(h,h2)
alias(k,k2)
alias(t,t2)
alias(p,p2)

Scalar HoraTrabajo Porcentaje de la hora que toma dar una materia /0.75/
Scalar MaxProfesor Maximo de horas de trabajo del profesor /35/
Scalar MaxDuracionMateria Maximo numero de veces que se puede dictar la materia al dia /2/

parameter ProfesorMateria(m,c,p) Si el profesor p dicta la materia m para el curso c;
$call GDXXRW Nuevo.xlsx par=ProfesorMateria rng=ProfesorMateria!a1:ba1000 dim=3 Cdim=1 Rdim=2
$GDGIN Nuevo.gdx

```

\$LOAD ProfesorMateria
\$GDXIN

parameter HorarioProfesor(d,h,p) Si el profesor p puede dictar clase el dia d en la hora h;
\$call GDXXRW Nuevo.xlsx par=HorarioProfesor rng=HorarioProfesor!a1:ba1000 dim=3 Cdim=1 Rdim=2
\$GDXIN Nuevo.gdx
\$LOAD HorarioProfesor
\$GDXIN

parameter HCursos(m,c) Numero de horas que la materia m se la debe dictar en el curso c;
\$call GDXXRW Nuevo.xlsx par=HCursos rng=HCursos!a1:ba1000 dim=2 Cdim=1 Rdim=1
\$GDXIN Nuevo.gdx
\$LOAD HCursos
\$GDXIN

parameter MateriasAula(m,t) Si la materia m se la dicta en el tipo de aula t;
\$call GDXXRW Nuevo.xlsx par=MateriasAula rng=MateriasAula!a1:ba1000 dim=2 Cdim=1 Rdim=1
\$GDXIN Nuevo.gdx
\$LOAD MateriasAula
\$GDXIN

parameter MateriaCursos(m,c) Si la materia m se la dicta en el curso c;
\$call GDXXRW Nuevo.xlsx par=MateriaCursos rng=MateriaCursos!a1:ba1000 dim=2 Cdim=1 Rdim=1
\$GDXIN Nuevo.gdx
\$LOAD MateriaCursos
\$GDXIN

parameter ExistenciaTipoAula(k,t) Si existe esa aula k y que tipo de aula es (t);
\$call GDXXRW Nuevo.xlsx par=ExistenciaTipoAula rng=ExistenciaTipoAula!a1:ba1000 dim=2 Cdim=1 Rdim=1
\$GDXIN Nuevo.gdx
\$LOAD ExistenciaTipoAula
\$GDXIN

parameter CursosParalelos(c,l) Si el curso c tiene el paralelo l abierto;
\$call GDXXRW Nuevo.xlsx par=CursosParalelos rng=CursosParalelos!a1:ba1000 dim=2 Cdim=1 Rdim=1
\$GDXIN Nuevo.gdx
\$LOAD CursosParalelos
\$GDXIN

set tupla(p,m,k,t,c,l,d,h) Conjunto de los casos que son factibles para reducir el tiempo de ejecucion ;
tupla(p,m,k,t,c,l,d,h)\$(ExistenciaTipoAula(k,t) and MateriasAula(m,t) and HCursos(m,c) and
HorarioProfesor(d,h,p) and ProfesorMateria(m,c,p) and CursosParalelos(c,l) and MateriaCursos(m,c))=yes;

variable

z Valor de la Funcion Objetivo

PAR(c,l,h,d) Matriz de curso paralelo por dia y hora

PROF(p,h,d) Matriz de horas de trabajo del profesor por dia y hora

AULA(k,t,d,h) Matriz de horas de ocupacion de las aulas tk por dia y hora

MATERIA(m,h,d) Matriz de la hora en que se dicta la materia m por dia y hora

HORARIO(m,c,l,d,h) Matriz del horario de la materia dictada en cada curso paralelo por dia y hora

binary variable

X(p,m,k,t,c,l,d,h) Variable del problema que asigna el horario para la escuela

equations

fo Funcion Objetivo

schedule Aqui se genera el horario de los cursos y sus paralelos

horalaboral Maximo de horas que un profesor puede trabajar

maximomateria Una materia debe durar maximo 2 horas en un dia

ocupacionaula Un aula solo puede ser ocupada por un profesor que dicta una materia en un curso paralelo en un dia y hora especifico

asignarclase Se asigna una clase con un profesor una materia en una aula para cada curso paralelo en cada hora y dia

asignarprofesor Se asigna a cada profesor cada hora y cada día a un máximo de un curso para lo que dicte una materia en una aula

flujo1 Flujo de una materia debe ser seguido
flujo2 Flujo seguido de una materia no puede ser saltado
flujo3 Flujo seguido de una materia no puede ser saltado
flujo4 Flujo seguido de una materia no puede ser saltado
flujo5 Flujo seguido de una materia no puede ser saltado
flujo6 Flujo seguido de una materia no puede ser saltado
flujo7 Flujo seguido de una materia no puede ser saltado
flujo8 Flujo seguido de una materia no puede ser saltado
flujo9 Flujo seguido de una materia no puede ser saltado

matriz1 Matriz que presenta una respuesta
matriz2 Matriz que presenta una respuesta
matriz3 Matriz que presenta una respuesta
matriz4 Matriz que presenta una respuesta
matriz5 Matriz que presenta una respuesta
;

fo.. z=e=sum[(p,m,k,t,c,l,d,h),X(p,m,k,t,c,l,d,h)];

schedule(m,c,l).. sum[tupla(p,m,k,t,c,l,d,h),X(p,m,k,t,c,l,d,h)]=g=HCursos(m,c);
horalaboral(p).. sum[(m,k,t,c,l,d,h),X(p,m,k,t,c,l,d,h)]*horatrabajo=I=Maxprofesor;
maximomateria(m,p,c,l,d).. Sum[tupla(p,m,k,t,c,l,d,h),X(p,m,k,t,c,l,d,h)]=I=MaxDuracionMateria;
ocupacionaula(t,k,d,h).. sum[tupla(p,m,k,t,c,l,d,h),X(p,m,k,t,c,l,d,h)]=I=1;
asignarclase(c,l,h,d).. Sum[tupla(p,m,k,t,c,l,d,h),X(p,m,k,t,c,l,d,h)]=I=1;
asignarprofesor(p,h,d).. Sum[tupla(p,m,k,t,c,l,d,h),X(p,m,k,t,c,l,d,h)]=I=1;

flujo1(tupla(p,m,k,t,c,l,d,h))\$(ord(h)<card(h)-1).. X(p,m,k,t,c,l,d,h)+X(p,m,k,t,c,l,d,h+1)=I=2;
flujo2(tupla(p,m,k,t,c,l,d,h))\$(ord(h)<card(h)-1).. X(p,m,k,t,c,l,d,h)+Sum[(p2,k2,t2)\$(ord(k2) ne
ord(k)),X(p2,m,k2,t2,c,l,d,h+1)]=I=1;
flujo3(tupla(p,m,k,t,c,l,d,h))\$(ord(h)<card(h)-2).. X(p,m,k,t,c,l,d,h)+sum[(p2,k2,t2),X(p2,m,k2,t2,c,l,d,h+2)]=I=1;
flujo4(tupla(p,m,k,t,c,l,d,h))\$(ord(h)<card(h)-3).. X(p,m,k,t,c,l,d,h)+sum[(p2,k2,t2),X(p2,m,k2,t2,c,l,d,h+3)]=I=1;
flujo5(tupla(p,m,k,t,c,l,d,h))\$(ord(h)<card(h)-4).. X(p,m,k,t,c,l,d,h)+sum[(p2,k2,t2),X(p2,m,k2,t2,c,l,d,h+4)]=I=1;
flujo6(tupla(p,m,k,t,c,l,d,h))\$(ord(h)<card(h)-5).. X(p,m,k,t,c,l,d,h)+sum[(p2,k2,t2),X(p2,m,k2,t2,c,l,d,h+5)]=I=1;
flujo7(tupla(p,m,k,t,c,l,d,h))\$(ord(h)<card(h)-6).. X(p,m,k,t,c,l,d,h)+sum[(p2,k2,t2),X(p2,m,k2,t2,c,l,d,h+6)]=I=1;
flujo8(tupla(p,m,k,t,c,l,d,h))\$(ord(h)<card(h)-7).. X(p,m,k,t,c,l,d,h)+sum[(p2,k2,t2),X(p2,m,k2,t2,c,l,d,h+7)]=I=1;
flujo9(tupla(p,m,k,t,c,l,d,h)).. X(p,m,k,t,c,l,d,'1')+sum[(k2,t2),X(p,m,k2,t2,c,l,d,'9')]=I=1;

matriz1(c,l,h,d).. Sum[(p,m,k,t),X(p,m,k,t,c,l,d,h)]=e=PAR(c,l,h,d);
matriz2(p,h,d).. Sum[(c,l,m,k,t),X(p,m,k,t,c,l,d,h)]=e=PROF(p,h,d);
matriz3(t,k,h,d).. Sum[(p,c,l,m),X(p,m,k,t,c,l,d,h)]=e=AULA(k,t,d,h);
matriz4(m,h,d).. Sum[(p,l,c,k,t),X(p,m,k,t,c,l,d,h)]=e=MATERIA(m,h,d);
matriz5(m,c,l,h,d).. Sum[(p,k,t),X(p,m,k,t,c,l,d,h)]=e=HORARIO(m,c,l,d,h);

model
prob /all/;

solve
prob using mip min z;

EXECUTE_UNLOAD "nuevo.gdx" par.l,prof.l,AULA.l,MATERIA.l,HORARIO.l,X.l
execute 'gdxxrw.exe nuevo.gdx var=par.l rng=Resultado1!A1:BZ1000 dim=4 Cdim=1 Rdim=3'
execute 'gdxxrw.exe nuevo.gdx var=prof.l rng=Resultado2!A1:BZ1000 dim=3 Cdim=1 Rdim=2'
execute 'gdxxrw.exe nuevo.gdx var=AULA.l rng=Resultado3!A1:BZ1000 dim=4 Cdim=2 Rdim=2'
execute 'gdxxrw.exe nuevo.gdx var=MATERIA.l rng=Resultado4!A1:BZ1000 dim=3 Cdim=1 Rdim=2'
execute 'gdxxrw.exe nuevo.gdx var=HORARIO.l rng=Resultado5!A1:BZ1000 dim=5 Cdim=2 Rdim=3'
execute 'gdxxrw.exe nuevo.gdx var=X.l rng=ResultadoX!A1:BZ1000 dim=8 Cdim=2 Rdim=6'

4. Base de Datos

MySQL 5.7 es la versión utilizada para el desarrollo del sistema

4.1 Credenciales

El usuario utilizado para este proyecto es admin el cual fue creado únicamente con los privilegios necesarios para el sistema es decir permisos sobre la base de datos correspondiente, mismo que será utilizado para la generación de reportes y la conexión del sistema con la base de datos.

4.2 Script de generación

```
CREATE DATABASE IF NOT EXISTS `gaha` /*!40100 DEFAULT CHARACTER SET utf8 */;
USE `gaha`;

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8 */;
/*!40103 SET @OLD_TIME_ZONE=@@TIME_ZONE */;
/*!40103 SET TIME_ZONE='+00:00' */;
/*!40014 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0 */;
/*!40014 SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0
*/;
/*!40101 SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;
/*!40111 SET @OLD_SQL_NOTES=@@SQL_NOTES, SQL_NOTES=0 */;

--
-- Table structure for table `arbolopciones`
--

DROP TABLE IF EXISTS `arbolopciones`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `arbolopciones` (
  `Padre` int(11) NOT NULL COMMENT 'Opción padre del menú',
  `Hijo` int(11) NOT NULL COMMENT 'Opción hijo del menú',
  `Profundidad` int(11) NOT NULL COMMENT 'Nivel de profundidad de la opción',
  PRIMARY KEY (`Padre`,`Hijo`,`Profundidad`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `aulas`
--

DROP TABLE IF EXISTS `aulas`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `aulas` (
  `ID_Aula` varchar(10) NOT NULL COMMENT 'Código del aula',
  `Aula` varchar(10) NOT NULL COMMENT 'Nombre del aula',
  `Asientos` int(11) NOT NULL COMMENT 'Cantidad de asientos',
```

```

`ID_TipoAula` int(11) NOT NULL COMMENT 'Código interno del tipo de aula',
`Comentarios` varchar(200) DEFAULT NULL COMMENT 'Comentario sobre el aula',
`ID_Estado` int(11) NOT NULL COMMENT 'Código interno del estado',
`FechaCreacion` datetime NOT NULL COMMENT 'Fecha de creación del aula',
`UsuarioCrea` int(11) NOT NULL COMMENT 'Código interno del usuario de creación del aula',
`FechaModificacion` datetime DEFAULT NULL COMMENT 'Fecha de modificación del aula',
`UsuarioModifica` int(11) DEFAULT NULL COMMENT 'Código interno del usuario de modificación del aula',
PRIMARY KEY (`ID_Aula`),
KEY `FK_ID_TipoAula2_idx` (`ID_TipoAula`),
KEY `FK_ID_Estado_idx` (`ID_Estado`),
CONSTRAINT `FK_ID_Estado` FOREIGN KEY (`ID_Estado`) REFERENCES `estados` (`ID_Estado`) ON
DELETE NO ACTION ON UPDATE NO ACTION,
CONSTRAINT `FK_ID_TipoAula2` FOREIGN KEY (`ID_TipoAula`) REFERENCES `tipopaula` (`ID_TipoAula`)
ON DELETE NO ACTION ON UPDATE NO ACTION
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
/*!40101 SET character_set_client = @saved_cs_client */;

```

```

--
-- Table structure for table `cursos`
--

```

```

DROP TABLE IF EXISTS `cursos`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `cursos` (
  `ID_Curso` varchar(5) NOT NULL COMMENT 'Código del curso',
  `Curso` varchar(20) NOT NULL COMMENT 'Nombre del curso',
  `Paralelos` int(11) NOT NULL COMMENT 'Cantidad de paralelos',
  `ID_Estado` int(11) NOT NULL COMMENT 'Código interno del estado',
  `FechaCreacion` datetime NOT NULL COMMENT 'Fecha de creación del curso',
  `UsuarioCrea` int(11) NOT NULL COMMENT 'Código interno del usuario de creación del curso',
  `FechaModificacion` datetime DEFAULT NULL COMMENT 'Fecha de modificación del curso',
  `UsuarioModifica` int(11) DEFAULT NULL COMMENT 'Código interno del usuario de modificación del curso',
PRIMARY KEY (`ID_Curso`),
KEY `FK_ID_Estado2_idx` (`ID_Estado`),
CONSTRAINT `FK_ID_Estado2` FOREIGN KEY (`ID_Estado`) REFERENCES `estados` (`ID_Estado`) ON
DELETE NO ACTION ON UPDATE NO ACTION
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
/*!40101 SET character_set_client = @saved_cs_client */;

```

```

--
-- Table structure for table `cursosparalelos`
--

```

```

DROP TABLE IF EXISTS `cursosparalelos`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `cursosparalelos` (
  `ID_Curso` varchar(5) NOT NULL,
  `ID_Paralelo` varchar(3) NOT NULL,
  `ID_Estado` int(11) NOT NULL,
  `FechaCreacion` datetime NOT NULL,
  `UsuarioCrea` int(11) NOT NULL,
  `FechaModificacion` datetime DEFAULT NULL,
  `UsuarioModifica` int(11) DEFAULT NULL,
PRIMARY KEY (`ID_Curso`,`ID_Paralelo`),
KEY `FK_ID_EstadoCursosParalelos_idx` (`ID_Estado`),
CONSTRAINT `FK_ID_EstadoCursosParalelos` FOREIGN KEY (`ID_Estado`) REFERENCES `estados`
(`ID_Estado`) ON DELETE NO ACTION ON UPDATE NO ACTION
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
/*!40101 SET character_set_client = @saved_cs_client */;

```

```

--
-- Table structure for table `dias`
--

DROP TABLE IF EXISTS `dias`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `dias` (
  `ID_Dia` int(11) NOT NULL AUTO_INCREMENT COMMENT 'Código interno del día',
  `Dia` varchar(10) NOT NULL COMMENT 'Nombre del día',
  `ID_Estado` int(11) NOT NULL COMMENT 'Código interno del estado',
  `FechaCreacion` datetime NOT NULL COMMENT 'Fecha de creación del día',
  `UsuarioCrea` int(11) NOT NULL COMMENT 'Código interno del usuario de creación del día',
  `FechaModificacion` datetime DEFAULT NULL COMMENT 'Fecha de modificación del día',
  `UsuarioModifica` int(11) DEFAULT NULL COMMENT 'Código interno del usuario de modificación del día',
  PRIMARY KEY (`ID_Dia`),
  KEY `FK_ID_Estado3_idx` (`ID_Estado`),
  CONSTRAINT `FK_ID_Estado3` FOREIGN KEY (`ID_Estado`) REFERENCES `estados` (`ID_Estado`) ON
DELETE NO ACTION ON UPDATE NO ACTION
) ENGINE=InnoDB AUTO_INCREMENT=8 DEFAULT CHARSET=utf8;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `dispprofesores`
--

DROP TABLE IF EXISTS `dispprofesores`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `dispprofesores` (
  `ID_Profesor` int(11) NOT NULL COMMENT 'Código interno del profesor',
  `ID_Dia` int(11) NOT NULL COMMENT 'Código interno del día',
  `ID_Hora` int(11) NOT NULL COMMENT 'Código interno de la hora',
  `ID_Estado` int(11) NOT NULL COMMENT 'Código interno del estado',
  `FechaCreacion` datetime NOT NULL COMMENT 'Fecha de creación de la disponibilidad del profesor',
  `UsuarioCrea` int(11) NOT NULL COMMENT 'Código interno del usuario de creación de la disponibilidad del
profesor',
  `FechaModificacion` datetime DEFAULT NULL COMMENT 'Fecha de modificación de la disponibilidad del
profesor',
  `UsuarioModifica` int(11) DEFAULT NULL COMMENT 'Código interno del usuario de modificación de la
disponibilidad del profesor',
  PRIMARY KEY (`ID_Profesor`,`ID_Dia`,`ID_Hora`),
  KEY `FK_ID_Profesor2_idx` (`ID_Profesor`),
  KEY `FK_ID_Hora2_idx` (`ID_Hora`),
  KEY `FK_ID_Dia_idx` (`ID_Dia`),
  KEY `FK_ID_Estado4_idx` (`ID_Estado`),
  CONSTRAINT `FK_ID_Dia` FOREIGN KEY (`ID_Dia`) REFERENCES `dias` (`ID_Dia`) ON DELETE NO
ACTION ON UPDATE NO ACTION,
  CONSTRAINT `FK_ID_Estado4` FOREIGN KEY (`ID_Estado`) REFERENCES `estados` (`ID_Estado`) ON
DELETE NO ACTION ON UPDATE NO ACTION,
  CONSTRAINT `FK_ID_Hora` FOREIGN KEY (`ID_Hora`) REFERENCES `horas` (`ID_Hora`) ON DELETE NO
ACTION ON UPDATE NO ACTION,
  CONSTRAINT `FK_ID_Profesor2` FOREIGN KEY (`ID_Profesor`) REFERENCES `profesores` (`ID_Profesor`)
ON DELETE NO ACTION ON UPDATE NO ACTION
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `estados`
--

```

```

DROP TABLE IF EXISTS `estados`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `estados` (
  `ID_Estado` int(11) NOT NULL AUTO_INCREMENT COMMENT 'Código interno del estado',
  `Estado` varchar(50) NOT NULL COMMENT 'Nombre del estado',
  `Descripcion` varchar(100) DEFAULT NULL COMMENT 'Descripción del estado',
  `FechaCreacion` datetime NOT NULL COMMENT 'Fecha de creación del estado',
  `UsuarioCrea` int(11) NOT NULL COMMENT 'Código interno del usuario de creación del estado',
  `FechaModificacion` datetime DEFAULT NULL COMMENT 'Fecha de modificación del estado',
  `UsuarioModifica` int(11) DEFAULT NULL COMMENT 'Código interno del usuario de modificación del estado',
  PRIMARY KEY (`ID_Estado`)
) ENGINE=InnoDB AUTO_INCREMENT=4 DEFAULT CHARSET=utf8;
/*!40101 SET character_set_client = @saved_cs_client */;

```

```

--
-- Table structure for table `horarioscabecera`
--

```

```

DROP TABLE IF EXISTS `horarioscabecera`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `horarioscabecera` (
  `ID_HorarioCab` int(11) NOT NULL AUTO_INCREMENT COMMENT 'Código interno de la cabecera del horario',
  `ID_SchollarYear` int(11) NOT NULL COMMENT 'Código interno del año escolar',
  `ID_Estado` int(11) NOT NULL COMMENT 'Código interno del estado',
  `FechaCreacion` datetime NOT NULL COMMENT 'Fecha de creación de la cabecera del horario',
  `UsuarioCrea` int(11) NOT NULL COMMENT 'Código interno del usuario de creación de la cabecera del horario',
  `FechaModificacion` datetime DEFAULT NULL COMMENT 'Fecha de modificación de la cabecera del horario',
  `UsuarioModifica` int(11) DEFAULT NULL COMMENT 'Código interno del usuario de modificación de la cabecera del horario',
  PRIMARY KEY (`ID_HorarioCab`),
  KEY `FK_ID_SchollarYear_idx` (`ID_SchollarYear`),
  KEY `FK_ID_Estado5_idx` (`ID_Estado`),
  CONSTRAINT `FK_ID_Estado5` FOREIGN KEY (`ID_Estado`) REFERENCES `estados` (`ID_Estado`) ON DELETE NO ACTION ON UPDATE NO ACTION,
  CONSTRAINT `FK_ID_SchollarYear` FOREIGN KEY (`ID_SchollarYear`) REFERENCES `schollaryear` (`ID_SchollarYear`) ON DELETE NO ACTION ON UPDATE NO ACTION
) ENGINE=InnoDB AUTO_INCREMENT=2 DEFAULT CHARSET=utf8;
/*!40101 SET character_set_client = @saved_cs_client */;

```

```

--
-- Table structure for table `horarioscabeceratmp`
--

```

```

DROP TABLE IF EXISTS `horarioscabeceratmp`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `horarioscabeceratmp` (
  `ID_HorarioCabTMP` int(11) NOT NULL AUTO_INCREMENT COMMENT 'Código interno de la cabecera del horario',
  `ID_SchollarYear` int(11) NOT NULL COMMENT 'Código interno del año escolar',
  `ID_Estado` int(11) NOT NULL COMMENT 'Código interno del estado',
  `FechaCreacion` datetime NOT NULL COMMENT 'Fecha de creación de la cabecera del horario',
  `UsuarioCrea` int(11) NOT NULL COMMENT 'Código interno del usuario de creación de la cabecera del horario',
  `FechaModificacion` datetime DEFAULT NULL COMMENT 'Fecha de modificación de la cabecera del horario',

```

```

`UsuarioModifica` int(11) DEFAULT NULL COMMENT 'Código interno del usuario de modificación de la
cabecera del horario',
PRIMARY KEY (`ID_HorarioCabTMP`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `horariosdetalle`
--

DROP TABLE IF EXISTS `horariosdetalle`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `horariosdetalle` (
  `ID_HorarioDet` int(11) NOT NULL AUTO_INCREMENT COMMENT 'Código interno del detalle del horario',
  `ID_HorarioCab` int(11) NOT NULL COMMENT 'Código interno de la cabecera del horario',
  `ID_Profesor` int(11) NOT NULL COMMENT 'Código interno del profesor',
  `ID_Materia` varchar(20) NOT NULL COMMENT 'Código de la materia',
  `ID_Curso` varchar(5) NOT NULL COMMENT 'Código interno del curso',
  `Paralelo` varchar(5) NOT NULL COMMENT 'Paralelo',
  `ID_Aula` varchar(10) NOT NULL COMMENT 'Código del aula',
  `ID_Dia` int(11) NOT NULL COMMENT 'Código interno del día',
  `ID_Hora` int(11) NOT NULL COMMENT 'Código interno del periodo de clases',
  `ID_Estado` int(11) NOT NULL COMMENT 'Código interno del estado',
  `FechaCreacion` datetime NOT NULL COMMENT 'Fecha de creación del detalle del horario',
  `UsuarioCrea` int(11) NOT NULL COMMENT 'Código interno del usuario de creación del detalle del horario',
  `FechaModificacion` datetime DEFAULT NULL COMMENT 'Fecha de modificación del detalle del horario',
  `UsuarioModifica` int(11) DEFAULT NULL COMMENT 'Código interno del usuario de modificación del detalle
del horario',
PRIMARY KEY (`ID_HorarioDet`,`ID_HorarioCab`),
KEY `FK_ID_ProfesoresMat_idx` (`ID_Profesor`,`ID_Materia`,`ID_Curso`),
KEY `FK_ID_Estado6_idx` (`ID_Estado`),
KEY `FK_ID_HorarioCab_idx` (`ID_HorarioCab`),
KEY `FK_ID_Dispprofesores_idx` (`ID_Profesor`,`ID_Dia`,`ID_Hora`),
KEY `FK_ID_Aula_idx` (`ID_Aula`),
CONSTRAINT `FK_ID_Aula` FOREIGN KEY (`ID_Aula`) REFERENCES `aulas` (`ID_Aula`) ON DELETE NO
ACTION ON UPDATE NO ACTION,
CONSTRAINT `FK_ID_Dispprofesores` FOREIGN KEY (`ID_Profesor`,`ID_Dia`,`ID_Hora`) REFERENCES
`dispprofesores` (`ID_Profesor`,`ID_Dia`,`ID_Hora`) ON DELETE NO ACTION ON UPDATE NO ACTION,
CONSTRAINT `FK_ID_Estado6` FOREIGN KEY (`ID_Estado`) REFERENCES `estados` (`ID_Estado`) ON
DELETE NO ACTION ON UPDATE NO ACTION,
CONSTRAINT `FK_ID_HorarioCab` FOREIGN KEY (`ID_HorarioCab`) REFERENCES `horarioscabecera`
(`ID_HorarioCab`) ON DELETE NO ACTION ON UPDATE NO ACTION,
CONSTRAINT `FK_ID_ProfesoresMat` FOREIGN KEY (`ID_Profesor`,`ID_Materia`,`ID_Curso`)
REFERENCES `profesoresmaterias` (`ID_Profesor`,`ID_Materia`,`ID_Curso`) ON DELETE NO ACTION ON
UPDATE NO ACTION
) ENGINE=InnoDB AUTO_INCREMENT=129 DEFAULT CHARSET=utf8;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `horariosdetalletmp`
--

DROP TABLE IF EXISTS `horariosdetalletmp`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `horariosdetalletmp` (
  `ID_HorariosDetTMP` int(11) NOT NULL AUTO_INCREMENT COMMENT 'Código interno del detalle del
horario',
  `ID_HorarioCabTMP` int(11) NOT NULL COMMENT 'Código interno de la cabecera del horario',
  `ID_Profesor` int(11) NOT NULL COMMENT 'Código interno del profesor',

```

```

`ID_Materia` varchar(20) NOT NULL COMMENT 'Código de la materia',
`ID_Curso` varchar(5) NOT NULL COMMENT 'Código interno del curso',
`Paralelo` varchar(5) NOT NULL COMMENT 'Paralelo',
`ID_Aula` varchar(10) NOT NULL COMMENT 'Código del aula',
`ID_Dia` varchar(5) NOT NULL COMMENT 'Código interno del día',
`ID_Hora` int(11) NOT NULL COMMENT 'Código interno del periodo de clases',
`ID_Estado` int(11) NOT NULL COMMENT 'Código interno del estado',
`FechaCreacion` datetime NOT NULL COMMENT 'Fecha de creación del detalle del horario',
`UsuarioCrea` int(11) NOT NULL COMMENT 'Código interno del usuario de creación del detalle del horario',
`FechaModificacion` datetime DEFAULT NULL COMMENT 'Fecha de modificación del detalle del horario',
`UsuarioModifica` int(11) DEFAULT NULL COMMENT 'Código interno del usuario de modificación del detalle
del horario',
PRIMARY KEY (`ID_HorariosDetTMP`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
/*!40101 SET character_set_client = @saved_cs_client */;

```

```

--
-- Table structure for table `horas`
--

```

```

DROP TABLE IF EXISTS `horas`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `horas` (
  `ID_Hora` int(11) NOT NULL AUTO_INCREMENT COMMENT 'Código interno del periodo de clases',
  `HoraIni` time(5) NOT NULL COMMENT 'Hora inicial del periodo de clases',
  `HoraFin` time(5) NOT NULL COMMENT 'Hora final del periodo de clases',
  `Comentarios` varchar(200) DEFAULT NULL COMMENT 'Comentarios sobre el periodo de clases',
  `ID_Estado` int(11) NOT NULL COMMENT 'Código interno del estado',
  `FechaCreacion` datetime NOT NULL COMMENT 'Fecha de creación del periodo de clases',
  `UsuarioCrea` int(11) NOT NULL COMMENT 'Código interno del usuario de creación del periodo de clases',
  `FechaModificacion` datetime DEFAULT NULL COMMENT 'Fecha de modificación del periodo de clases',
  `UsuarioModifica` int(11) DEFAULT NULL COMMENT 'Código interno del usuario de modificación del periodo
de clases',
PRIMARY KEY (`ID_Hora`),
KEY `FK_ID_Estado7_idx` (`ID_Estado`),
CONSTRAINT `FK_ID_Estado7` FOREIGN KEY (`ID_Estado`) REFERENCES `estados` (`ID_Estado`) ON
DELETE NO ACTION ON UPDATE NO ACTION
) ENGINE=InnoDB AUTO_INCREMENT=12 DEFAULT CHARSET=utf8;
/*!40101 SET character_set_client = @saved_cs_client */;

```

```

--
-- Table structure for table `materias`
--

```

```

DROP TABLE IF EXISTS `materias`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `materias` (
  `ID_Materia` varchar(20) NOT NULL COMMENT 'Código de materia',
  `Materia` varchar(100) NOT NULL COMMENT 'Nombre de la materia',
  `ID_TipoAula` int(11) NOT NULL COMMENT 'Código interno del tipo de aula',
  `ID_Estado` int(11) NOT NULL COMMENT 'Código interno del estado',
  `FechaCreacion` datetime NOT NULL COMMENT 'Fecha de creación de la materia',
  `UsuarioCrea` int(11) NOT NULL COMMENT 'Código interno del usuario de creación de la materia',
  `FechaModificacion` datetime DEFAULT NULL COMMENT 'Fecha de modificación de la materia',
  `UsuarioModifica` int(11) DEFAULT NULL COMMENT 'Código interno del usuario de modificación de la
materia',
PRIMARY KEY (`ID_Materia`),
KEY `FK_ID_TipoAula_idx` (`ID_TipoAula`),
KEY `FK_ID_Estado8_idx` (`ID_Estado`),

```

```

CONSTRAINT `FK_ID_Estado8` FOREIGN KEY (`ID_Estado`) REFERENCES `estados` (`ID_Estado`) ON
DELETE NO ACTION ON UPDATE NO ACTION,
CONSTRAINT `FK_ID_TipoAula` FOREIGN KEY (`ID_TipoAula`) REFERENCES `tipoaula` (`ID_TipoAula`) ON
DELETE NO ACTION ON UPDATE NO ACTION
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
/*!40101 SET character_set_client = @saved_cs_client */;

```

```

--
-- Table structure for table `materiascursos`
--

```

```

DROP TABLE IF EXISTS `materiascursos`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `materiascursos` (
  `ID_Materia` varchar(20) NOT NULL COMMENT 'Código de la materia',
  `ID_Curso` varchar(5) NOT NULL COMMENT 'Código del curso',
  `HorasSemana` int(11) NOT NULL COMMENT 'Horas a la semana de la materia por curso',
  `ID_Estado` int(11) NOT NULL COMMENT 'Código interno del estado',
  `FechaCreacion` datetime NOT NULL COMMENT 'Fecha de creación de la materia por curso',
  `UsuarioCrea` int(11) NOT NULL COMMENT 'Código interno del usuario de creación de la materia por curso',
  `FechaModificacion` datetime DEFAULT NULL COMMENT 'Fecha de modificación de la materia por curso',
  `UsuarioModifica` int(11) DEFAULT NULL COMMENT 'Código interno del usuario de modificación de la mateira
por curso',
  PRIMARY KEY (`ID_Materia`,`ID_Curso`),
  KEY `FK_ID_Curso_idx` (`ID_Curso`),
  KEY `FK_ID_Estado9_idx` (`ID_Estado`),
  CONSTRAINT `FK_ID_Curso` FOREIGN KEY (`ID_Curso`) REFERENCES `cursos` (`ID_Curso`) ON DELETE
NO ACTION ON UPDATE NO ACTION,
  CONSTRAINT `FK_ID_Estado9` FOREIGN KEY (`ID_Estado`) REFERENCES `estados` (`ID_Estado`) ON
DELETE NO ACTION ON UPDATE NO ACTION,
  CONSTRAINT `FK_ID_Materia` FOREIGN KEY (`ID_Materia`) REFERENCES `materias` (`ID_Materia`) ON
DELETE NO ACTION ON UPDATE NO ACTION
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
/*!40101 SET character_set_client = @saved_cs_client */;

```

```

--
-- Table structure for table `opciones`
--

```

```

DROP TABLE IF EXISTS `opciones`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `opciones` (
  `ID_Opcion` int(11) NOT NULL AUTO_INCREMENT COMMENT 'Código interno de la opción',
  `Opcion` varchar(100) NOT NULL COMMENT 'Nombre de la opción',
  `OpcionContenedora` int(11) DEFAULT NULL COMMENT 'Código interno de la opción contenedora',
  `Orden` tinytext NOT NULL COMMENT 'Orden de la opción',
  `TituloMenu` varchar(50) NOT NULL COMMENT 'Título de menú de la opción',
  `Imagen` varchar(150) DEFAULT NULL COMMENT 'Imagen de la opción',
  `URL` varchar(150) DEFAULT NULL COMMENT 'Dirección url de la opción',
  `ID_Permiso` int(11) DEFAULT NULL COMMENT 'Código interno del permiso de la opción',
  `ID_Estado` int(11) NOT NULL COMMENT 'Código interno del estado',
  `FechaCreacion` datetime NOT NULL COMMENT 'Fecha de creación de la opción',
  `UsuarioCrea` int(11) NOT NULL COMMENT 'Código interno del usuario de creación de la opción',
  `FechaModificacion` datetime DEFAULT NULL COMMENT 'Fecha de modificación de la opción',
  `UsuarioModifica` int(11) DEFAULT NULL COMMENT 'Código interno del usuario de modificación de la opción',
  PRIMARY KEY (`ID_Opcion`),
  KEY `FK_ID_Permiso2_idx` (`ID_Permiso`),
  KEY `FK_ID_Estado10_idx` (`ID_Estado`),

```

```

    CONSTRAINT `FK_ID_Estado10` FOREIGN KEY (`ID_Estado`) REFERENCES `estados` (`ID_Estado`) ON
DELETE NO ACTION ON UPDATE NO ACTION,
    CONSTRAINT `FK_ID_Permito2` FOREIGN KEY (`ID_Permito`) REFERENCES `permisos` (`ID_Permito`) ON
DELETE NO ACTION ON UPDATE NO ACTION
) ENGINE=InnoDB AUTO_INCREMENT=22 DEFAULT CHARSET=utf8;
/*!40101 SET character_set_client = @saved_cs_client */;

```

```

--
-- Table structure for table `paralelos`
--

```

```

DROP TABLE IF EXISTS `paralelos`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `paralelos` (
  `ID_Paralelo` varchar(3) NOT NULL,
  `Paralelo` varchar(10) NOT NULL,
  `ID_Estado` int(11) NOT NULL,
  `FechaCreacion` datetime NOT NULL,
  `UsuarioCrea` int(11) NOT NULL,
  `FechaModificacion` datetime DEFAULT NULL,
  `UsuarioModifica` int(11) DEFAULT NULL,
  PRIMARY KEY (`ID_Paralelo`),
  KEY `FK_ID_EstadoParalelo_idx` (`ID_Estado`),
  CONSTRAINT `FK_ID_EstadoParalelo` FOREIGN KEY (`ID_Estado`) REFERENCES `estados` (`ID_Estado`)
ON DELETE NO ACTION ON UPDATE NO ACTION
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
/*!40101 SET character_set_client = @saved_cs_client */;

```

```

--
-- Table structure for table `perfilesusuario`
--

```

```

DROP TABLE IF EXISTS `perfilesusuario`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `perfilesusuario` (
  `ID_PerfilUsuario` int(11) NOT NULL AUTO_INCREMENT COMMENT 'Código interno del perfil de usuario',
  `Perfil` varchar(60) NOT NULL COMMENT 'Nombre del perfil de usuario',
  `Comentarios` varchar(200) DEFAULT NULL COMMENT 'Comentarios del perfil de usuario',
  `ID_Estado` int(11) NOT NULL COMMENT 'Código interno del estado',
  `FechaCreacion` datetime NOT NULL COMMENT 'Fecha de creación del perfil de usuario',
  `UsuarioCrea` int(11) NOT NULL COMMENT 'Código interno del usuario de creación del perfil de usuario',
  `FechaModificacion` datetime DEFAULT NULL COMMENT 'Fecha de modificación del perfil de usuario',
  `UsuarioModifica` int(11) DEFAULT NULL COMMENT 'Código interno del usuario de modificación del perfil de
usuario',
  PRIMARY KEY (`ID_PerfilUsuario`),
  KEY `FK_ID_Estado11_idx` (`ID_Estado`),
  CONSTRAINT `FK_ID_Estado11` FOREIGN KEY (`ID_Estado`) REFERENCES `estados` (`ID_Estado`) ON
DELETE NO ACTION ON UPDATE NO ACTION
) ENGINE=InnoDB AUTO_INCREMENT=2 DEFAULT CHARSET=utf8;
/*!40101 SET character_set_client = @saved_cs_client */;

```

```

--
-- Table structure for table `permisos`
--

```

```

DROP TABLE IF EXISTS `permisos`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `permisos` (

```

```

`ID_Permito` int(11) NOT NULL AUTO_INCREMENT COMMENT 'Código interno del permiso',
`Permito` varchar(60) NOT NULL COMMENT 'Nombre del permiso',
`ID_Estado` int(11) NOT NULL COMMENT 'Código interno del estado',
`FechaCreacion` datetime NOT NULL COMMENT 'Fecha de creación del permiso',
`UsuarioCrea` int(11) NOT NULL COMMENT 'Código interno del usuario de creación del permiso',
`FechaModificacion` datetime DEFAULT NULL COMMENT 'Fecha de modificación del permiso',
`UsuarioModifica` int(11) DEFAULT NULL COMMENT 'Código interno del usuario de modificación del permiso',
PRIMARY KEY (`ID_Permito`),
KEY `FK_ID_Estado12_idx` (`ID_Estado`),
CONSTRAINT `FK_ID_Estado12` FOREIGN KEY (`ID_Estado`) REFERENCES `estados` (`ID_Estado`) ON
DELETE NO ACTION ON UPDATE NO ACTION
) ENGINE=InnoDB AUTO_INCREMENT=22 DEFAULT CHARSET=utf8;
/*!40101 SET character_set_client = @saved_cs_client */;

```

```

--
-- Table structure for table `permisosperfiles`
--

```

```

DROP TABLE IF EXISTS `permisosperfiles`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `permisosperfiles` (
  `ID_Perfil` int(11) NOT NULL COMMENT 'Código interno del perfil de usuario',
  `ID_Permito` int(11) NOT NULL COMMENT 'Código interno del permiso',
  `ID_Estado` int(11) NOT NULL COMMENT 'Código interno del estado',
  `FechaCreacion` datetime NOT NULL COMMENT 'Fecha de creación de los permisos del perfil',
  `UsuarioCrea` int(11) NOT NULL COMMENT 'Código interno del usuario de creación de los permisos del perfil',
  `FechaModificacion` datetime DEFAULT NULL COMMENT 'Fecha de modificación de los permisos del perfil',
  `UsuarioModifica` int(11) DEFAULT NULL COMMENT 'Código interno del usuario de modificación de los
permisos del perfil',
PRIMARY KEY (`ID_Perfil`,`ID_Permito`),
KEY `FK_ID_Permito_idx` (`ID_Permito`),
KEY `FK_ID_Perfil_idx` (`ID_Perfil`),
KEY `FK_ID_Estado13_idx` (`ID_Estado`),
CONSTRAINT `FK_ID_Estado13` FOREIGN KEY (`ID_Estado`) REFERENCES `estados` (`ID_Estado`) ON
DELETE NO ACTION ON UPDATE NO ACTION,
CONSTRAINT `FK_ID_Perfil` FOREIGN KEY (`ID_Perfil`) REFERENCES `perfilesusuario` (`ID_PerfilUsuario`)
ON DELETE NO ACTION ON UPDATE NO ACTION,
CONSTRAINT `FK_ID_Permito` FOREIGN KEY (`ID_Permito`) REFERENCES `permisos` (`ID_Permito`) ON
DELETE NO ACTION ON UPDATE NO ACTION
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
/*!40101 SET character_set_client = @saved_cs_client */;

```

```

--
-- Table structure for table `profesores`
--

```

```

DROP TABLE IF EXISTS `profesores`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `profesores` (
  `ID_Profesor` int(11) NOT NULL AUTO_INCREMENT COMMENT 'Código interno del profesor',
  `Profesor` varchar(100) NOT NULL COMMENT 'Nombre del profesor',
  `Titulo` varchar(100) DEFAULT NULL COMMENT 'Título del profesor',
  `ID_Estado` int(11) NOT NULL COMMENT 'Código interno del estado',
  `FechaCreacion` datetime NOT NULL COMMENT 'Fecha de creación del profesor',
  `UsuarioCrea` int(11) NOT NULL COMMENT 'Código interno del usuario de creación del profesor',
  `FechaModificacion` datetime DEFAULT NULL COMMENT 'Fecha de modificación del profesor',
  `UsuarioModifica` int(11) DEFAULT NULL COMMENT 'Código interno del usuario de modificación del profesor',
PRIMARY KEY (`ID_Profesor`),
KEY `FK_ID_Estado14_idx` (`ID_Estado`),

```

```

CONSTRAINT `FK_ID_Estado14` FOREIGN KEY (`ID_Estado`) REFERENCES `estados` (`ID_Estado`) ON
DELETE NO ACTION ON UPDATE NO ACTION
) ENGINE=InnoDB AUTO_INCREMENT=19 DEFAULT CHARSET=utf8;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `profesoresmaterias`
--

DROP TABLE IF EXISTS `profesoresmaterias`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `profesoresmaterias` (
  `ID_Profesor` int(11) NOT NULL COMMENT 'Código interno del profesor',
  `ID_Materia` varchar(20) NOT NULL COMMENT 'Código de la materia',
  `ID_Curso` varchar(5) NOT NULL COMMENT 'Código del curso',
  `ID_Estado` int(11) NOT NULL COMMENT 'Código interno del estado',
  `FechaCreacion` datetime NOT NULL COMMENT 'Fecha de creación de las materias por curso del profesor',
  `UsuarioCrea` int(11) NOT NULL COMMENT 'Código interno del usuario de creación de las materias por curso
del profesor',
  `FechaModificacion` datetime DEFAULT NULL COMMENT 'Fecha de modificación de las materias por curso del
profesor',
  `UsuarioModifica` int(11) DEFAULT NULL COMMENT 'Código interno del usuario de modificación de las
materias por curso del profesor',
  PRIMARY KEY (`ID_Profesor`,`ID_Materia`,`ID_Curso`),
  KEY `FK_ID_Profesor3_idx` (`ID_Profesor`),
  KEY `FK_ID_MateriaCurso_idx` (`ID_Materia`,`ID_Curso`),
  KEY `FK_ID_Estado15_idx` (`ID_Estado`),
  CONSTRAINT `FK_ID_Estado15` FOREIGN KEY (`ID_Estado`) REFERENCES `estados` (`ID_Estado`) ON
DELETE NO ACTION ON UPDATE NO ACTION,
  CONSTRAINT `FK_ID_MateriaCurso` FOREIGN KEY (`ID_Materia`,`ID_Curso`) REFERENCES
`materiascursos` (`ID_Materia`,`ID_Curso`) ON DELETE NO ACTION ON UPDATE NO ACTION,
  CONSTRAINT `FK_ID_Profesor3` FOREIGN KEY (`ID_Profesor`) REFERENCES `profesores` (`ID_Profesor`)
ON DELETE NO ACTION ON UPDATE NO ACTION
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `schollaryear`
--

DROP TABLE IF EXISTS `schollaryear`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `schollaryear` (
  `ID_ScholarYear` int(11) NOT NULL AUTO_INCREMENT COMMENT 'Código interno del año lectivo',
  `SchollarYear` varchar(20) NOT NULL COMMENT 'Año lectivo',
  `FechaInicio` date NOT NULL COMMENT 'Fecha de inicio del año lectivo',
  `FechaFin` date NOT NULL COMMENT 'Fecha de fin del año lectivo',
  `DuracionClase` time(5) NOT NULL COMMENT 'Duración de las clases',
  `MaxHorasSemanaProfesor` int(11) NOT NULL COMMENT 'Máximo de horas a la semana del profesor',
  `ID_Estado` int(11) NOT NULL COMMENT 'Código interno del estado',
  `FechaCreacion` datetime NOT NULL COMMENT 'Fecha de creación del año lectivo',
  `UsuarioCrea` int(11) NOT NULL COMMENT 'Código interno del usuario de creación del año lectivo',
  `FechaModificacion` datetime DEFAULT NULL COMMENT 'Fecha de modificación del año lectivo',
  `UsuarioModifica` int(11) DEFAULT NULL COMMENT 'Código interno del usuario de modificación del año
lectivo',
  PRIMARY KEY (`ID_ScholarYear`),
  KEY `FK_ID_Estado16_idx` (`ID_Estado`),
  CONSTRAINT `FK_ID_Estado16` FOREIGN KEY (`ID_Estado`) REFERENCES `estados` (`ID_Estado`) ON
DELETE NO ACTION ON UPDATE NO ACTION

```

```

) ENGINE=InnoDB AUTO_INCREMENT=2 DEFAULT CHARSET=utf8;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `tipoaula`
--

DROP TABLE IF EXISTS `tipoaula`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `tipoaula` (
  `ID_TipoAula` int(11) NOT NULL AUTO_INCREMENT COMMENT 'Código interno del tipo de aula',
  `TipoAula` varchar(100) NOT NULL COMMENT 'Nombre del tipo de aula',
  `ID_Estado` int(11) NOT NULL COMMENT 'Código interno del estado',
  `FechaCreacion` datetime NOT NULL COMMENT 'Fecha de creación del tipo de aula',
  `UsuarioCrea` int(11) NOT NULL COMMENT 'Código interno del usuario de creación del tipo de aula',
  `FechaModificacion` datetime DEFAULT NULL COMMENT 'Fecha de modificación del tipo de aula',
  `UsuarioModifica` int(11) DEFAULT NULL COMMENT 'Código interno del usuario de modificación del tipo de aula',
  PRIMARY KEY (`ID_TipoAula`),
  KEY `FK_ID_Estado17_idx` (`ID_Estado`),
  CONSTRAINT `FK_ID_Estado17` FOREIGN KEY (`ID_Estado`) REFERENCES `estados` (`ID_Estado`) ON DELETE NO ACTION ON UPDATE NO ACTION
) ENGINE=InnoDB AUTO_INCREMENT=4 DEFAULT CHARSET=utf8;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `usuarios`
--

DROP TABLE IF EXISTS `usuarios`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `usuarios` (
  `ID_Usuario` int(11) NOT NULL AUTO_INCREMENT COMMENT 'Código interno del usuario',
  `Usuario` varchar(15) NOT NULL COMMENT 'Nombre de usuario',
  `Password` varchar(256) DEFAULT NULL COMMENT 'Contraseña',
  `Nombres` varchar(50) NOT NULL COMMENT 'Nombres',
  `Apellidos` varchar(50) NOT NULL COMMENT 'Apellidos',
  `Correo` varchar(70) DEFAULT NULL COMMENT 'Correo electrónico',
  `ID_PerfilUsuario` int(11) NOT NULL COMMENT 'Código interno del perfil de usuario',
  `ID_Estado` int(11) NOT NULL COMMENT 'Código interno del estado',
  `FechaCreacion` datetime NOT NULL COMMENT 'Fecha de creación del usuario',
  `UsuarioCrea` int(11) NOT NULL COMMENT 'Código interno del usuario de creación del usuario',
  `FechaModificacion` datetime DEFAULT NULL COMMENT 'Fecha de modificación del usuario',
  `UsuarioModifica` int(11) DEFAULT NULL COMMENT 'Código interno del usuario de modificación del usuario',
  PRIMARY KEY (`ID_Usuario`),
  UNIQUE KEY `Usuario_UNIQUE` (`Usuario`),
  KEY `FK_ID_PerfilUsuario_idx` (`ID_PerfilUsuario`),
  KEY `FK_ID_Estado18_idx` (`ID_Estado`),
  CONSTRAINT `FK_ID_Estado18` FOREIGN KEY (`ID_Estado`) REFERENCES `estados` (`ID_Estado`) ON DELETE NO ACTION ON UPDATE NO ACTION,
  CONSTRAINT `FK_ID_PerfilUsuario` FOREIGN KEY (`ID_PerfilUsuario`) REFERENCES `perfilesusuario` (`ID_PerfilUsuario`) ON DELETE NO ACTION ON UPDATE NO ACTION
) ENGINE=InnoDB AUTO_INCREMENT=2 DEFAULT CHARSET=utf8;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping routines for database 'gaha'
--
/*!50003 DROP FUNCTION IF EXISTS `menus` */;

```

```

/*!50003 SET @saved_cs_client      = @@character_set_client */ ;
/*!50003 SET @saved_cs_results     = @@character_set_results */ ;
/*!50003 SET @saved_col_connection = @@collation_connection */ ;
/*!50003 SET character_set_client  = utf8 */ ;
/*!50003 SET character_set_results = utf8 */ ;
/*!50003 SET collation_connection  = utf8_general_ci */ ;
/*!50003 SET @saved_sql_mode       = @@sql_mode */ ;
/*!50003 SET sql_mode               =
'STRICT_TRANS_TABLES,NO_AUTO_CREATE_USER,NO_ENGINE_SUBSTITUTION' */ ;
DELIMITER ;;
CREATE DEFINER='root'@'localhost' FUNCTION `menus`(menus_contenedores VARCHAR(100)) RETURNS
varchar(100) CHARSET utf8
BEGIN
    DECLARE result VARCHAR(100);

    SELECT
    CONCAT(GROUP_CONCAT(DISTINCT ID_Opcion),
    ',',
    CONCAT('\n',
    GROUP_CONCAT(DISTINCT CASE
    WHEN OpcionContenedora IS NULL THEN ID_Opcion
    ELSE OpcionContenedora
    END),
    '\n'))
    INTO result FROM
    opciones op
    WHERE
    FIND_IN_SET(ID_Opcion, menus_contenedores);

    RETURN result;
END ;;
DELIMITER ;
/*!50003 SET sql_mode              = @saved_sql_mode */ ;
/*!50003 SET character_set_client   = @saved_cs_client */ ;
/*!50003 SET character_set_results = @saved_cs_results */ ;
/*!50003 SET collation_connection  = @saved_col_connection */ ;
/*!50003 DROP FUNCTION IF EXISTS `SPLIT_STR` */ ;
/*!50003 SET @saved_cs_client      = @@character_set_client */ ;
/*!50003 SET @saved_cs_results     = @@character_set_results */ ;
/*!50003 SET @saved_col_connection = @@collation_connection */ ;
/*!50003 SET character_set_client  = utf8 */ ;
/*!50003 SET character_set_results = utf8 */ ;
/*!50003 SET collation_connection  = utf8_general_ci */ ;
/*!50003 SET @saved_sql_mode       = @@sql_mode */ ;
/*!50003 SET sql_mode               =
'STRICT_TRANS_TABLES,NO_AUTO_CREATE_USER,NO_ENGINE_SUBSTITUTION' */ ;
DELIMITER ;;
CREATE DEFINER='root'@'localhost' FUNCTION `SPLIT_STR`(
x VARCHAR(255),
delim VARCHAR(12),
pos INT
) RETURNS varchar(255) CHARSET utf8
RETURN REPLACE(SUBSTRING(SUBSTRING_INDEX(x, delim, pos),
LENGTH(SUBSTRING_INDEX(x, delim, pos-1)) + 1),
delim, "");
DELIMITER ;
/*!50003 SET sql_mode              = @saved_sql_mode */ ;
/*!50003 SET character_set_client   = @saved_cs_client */ ;
/*!50003 SET character_set_results = @saved_cs_results */ ;
/*!50003 SET collation_connection  = @saved_col_connection */ ;
/*!50003 DROP FUNCTION IF EXISTS `valida_login` */ ;

```

```

/*!50003 SET @saved_cs_client      = @@character_set_client */ ;
/*!50003 SET @saved_cs_results     = @@character_set_results */ ;
/*!50003 SET @saved_col_connection = @@collation_connection */ ;
/*!50003 SET character_set_client  = utf8 */ ;
/*!50003 SET character_set_results = utf8 */ ;
/*!50003 SET collation_connection  = utf8_general_ci */ ;
/*!50003 SET @saved_sql_mode       = @@sql_mode */ ;
/*!50003 SET sql_mode              =
'STRICT_TRANS_TABLES,NO_AUTO_CREATE_USER,NO_ENGINE_SUBSTITUTION' */ ;
DELIMITER ;;
CREATE DEFINER='root'@'localhost' FUNCTION `valida_login`(usuario varchar(15), pwd varchar(256))
RETURNS int(11)
BEGIN
    DECLARE idUsuario INT;
    SET idUsuario = 0;

SELECT
    a.ID_Usuario
INTO idUsuario FROM
    usuarios AS a
WHERE
    a.Usuario = usuario
    AND a.Password = SHA1(pwd)
    AND a.ID_Estado = 1;

    RETURN idUsuario;
END ;;
DELIMITER ;
/*!50003 SET sql_mode              = @saved_sql_mode */ ;
/*!50003 SET character_set_client  = @saved_cs_client */ ;
/*!50003 SET character_set_results = @saved_cs_results */ ;
/*!50003 SET collation_connection  = @saved_col_connection */ ;
/*!50003 DROP PROCEDURE IF EXISTS `arbol_menus` */;
/*!50003 SET @saved_cs_client      = @@character_set_client */ ;
/*!50003 SET @saved_cs_results     = @@character_set_results */ ;
/*!50003 SET @saved_col_connection = @@collation_connection */ ;
/*!50003 SET character_set_client  = utf8 */ ;
/*!50003 SET character_set_results = utf8 */ ;
/*!50003 SET collation_connection  = utf8_general_ci */ ;
/*!50003 SET @saved_sql_mode       = @@sql_mode */ ;
/*!50003 SET sql_mode              =
'STRICT_TRANS_TABLES,NO_AUTO_CREATE_USER,NO_ENGINE_SUBSTITUTION' */ ;
DELIMITER ;;
CREATE DEFINER='root'@'localhost' PROCEDURE `arbol_menus`(usuario VARCHAR(15))
BEGIN
    SELECT DISTINCT o.*
    FROM
        gaha.usuarios u
        INNER JOIN
        gaha.perfilesusuario pu ON u.ID_PerfilUsuario = pu.ID_PerfilUsuario
        INNER JOIN
        gaha.permisosperfiles pp ON pu.ID_PerfilUsuario = pp.ID_Perfil
        INNER JOIN
        gaha.opciones o ON pp.ID_Perfil = o.ID_Perfil
        JOIN
        gaha.arbolopciones ao ON o.ID_Opcion = ao.Hijo
    WHERE
        u.Usuario = usuario AND u.ID_Estado = 1
        AND pu.ID_Estado = 1
        AND pp.ID_Estado = 1
    AND ao.Profundidad = 0

```

```

ORDER BY o.Orden ASC;
END ;;
DELIMITER ;
/*!50003 SET sql_mode            = @saved_sql_mode */;
/*!50003 SET character_set_client = @saved_cs_client */;
/*!50003 SET character_set_results = @saved_cs_results */;
/*!50003 SET collation_connection = @saved_col_connection */;
/*!50003 DROP PROCEDURE IF EXISTS `crear_usuario` */;
/*!50003 SET @saved_cs_client     = @@character_set_client */;
/*!50003 SET @saved_cs_results   = @@character_set_results */;
/*!50003 SET @saved_col_connection = @@collation_connection */;
/*!50003 SET character_set_client = utf8 */;
/*!50003 SET character_set_results = utf8 */;
/*!50003 SET collation_connection = utf8_general_ci */;
/*!50003 SET @saved_sql_mode     = @@sql_mode */;
/*!50003 SET sql_mode            =
'STRICT_TRANS_TABLES,NO_AUTO_CREATE_USER,NO_ENGINE_SUBSTITUTION' */;
DELIMITER ;;
CREATE DEFINER='root'@`localhost` PROCEDURE `crear_usuario`(usuario varchar(15), pwd varchar(256),
nombres varchar(35), apellidos varchar(35), correo varchar(70), perfil int, usuarioCrea int)
BEGIN
    INSERT INTO `gaha`.`usuarios`
    (`Usuario`,
    `Password`,
    `Nombres`,
    `Apellidos`,
    `Correo`,
    `ID_PerfilUsuario`,
    `Estado`,
    `FechaCreacion`,
    `UsuarioCrea`)
    VALUES
    (usuario,
    sha1(pwd),
    nombres,
    apellidos,
    correo,
    perfil,
    1,
    sysdate(),
    usuarioCrea);

END ;;
DELIMITER ;
/*!50003 SET sql_mode            = @saved_sql_mode */;
/*!50003 SET character_set_client = @saved_cs_client */;
/*!50003 SET character_set_results = @saved_cs_results */;
/*!50003 SET collation_connection = @saved_col_connection */;
/*!50003 DROP PROCEDURE IF EXISTS `CursosParalelos` */;
/*!50003 SET @saved_cs_client     = @@character_set_client */;
/*!50003 SET @saved_cs_results   = @@character_set_results */;
/*!50003 SET @saved_col_connection = @@collation_connection */;
/*!50003 SET character_set_client = utf8 */;
/*!50003 SET character_set_results = utf8 */;
/*!50003 SET collation_connection = utf8_general_ci */;
/*!50003 SET @saved_sql_mode     = @@sql_mode */;
/*!50003 SET sql_mode            =
'STRICT_TRANS_TABLES,NO_AUTO_CREATE_USER,NO_ENGINE_SUBSTITUTION' */;
DELIMITER ;;
CREATE DEFINER='root'@`localhost` PROCEDURE `CursosParalelos`()
BEGIN

```

```

        SET @SQL = NULL;
SELECT
    GROUP_CONCAT(DISTINCT CONCAT('case when sum(case when r.id_paralelo = \'',
        dt,
        '\ then 1 else 0 end) > 0 then 1 else 0 end AS `',
        dt,
        '`'))
INTO @SQL FROM
    (SELECT
        d1.ID_Paralelo AS dt
    FROM
        paralelos d1
    ORDER BY d1.ID_Paralelo) d;

SET @SQL
= CONCAT('SELECT r.id_curso AS id_curso, ', @SQL, '
    from cursosparalelos r,
        paralelos s
    where r.id_paralelo = s.id_paralelo
    group by r.id_curso
    ');

PREPARE stmt FROM @SQL;
EXECUTE stmt;
DEALLOCATE PREPARE stmt;
END ;;
DELIMITER ;
/*!50003 SET sql_mode            = @saved_sql_mode */ ;
/*!50003 SET character_set_client = @saved_cs_client */ ;
/*!50003 SET character_set_results = @saved_cs_results */ ;
/*!50003 SET collation_connection = @saved_col_connection */ ;
/*!50003 DROP PROCEDURE IF EXISTS `ExistenciaTipoAula` */;
/*!50003 SET @saved_cs_client      = @@character_set_client */ ;
/*!50003 SET @saved_cs_results     = @@character_set_results */ ;
/*!50003 SET @saved_col_connection = @@collation_connection */ ;
/*!50003 SET character_set_client  = utf8 */ ;
/*!50003 SET character_set_results = utf8 */ ;
/*!50003 SET collation_connection  = utf8_general_ci */ ;
/*!50003 SET @saved_sql_mode       = @@sql_mode */ ;
/*!50003 SET sql_mode              =
'STRICT_TRANS_TABLES,NO_AUTO_CREATE_USER,NO_ENGINE_SUBSTITUTION' */ ;
DELIMITER ;;
CREATE DEFINER='root'@'localhost' PROCEDURE `ExistenciaTipoAula`()
BEGIN
    SET @SQL = NULL;
SELECT
    GROUP_CONCAT(DISTINCT CONCAT('sum(case when r.id_tipoaula = \'',
        dt,
        '\ then 1 else 0 end) AS `',
        dt,
        '`'))
INTO @SQL FROM
    (SELECT
        d1.ID_TipoAula AS dt
    FROM
        tipoaula d1
    ORDER BY d1.ID_TipoAula) d;

SET @SQL
= CONCAT('SELECT r.id_aula AS id_aula, ', @SQL, '
    from aulas r,

```

```

                                tipoaula s
      where r.id_tipoaula = s.id_tipoaula
      group by r.id_aula
      ');

PREPARE stmt FROM @SQL;
EXECUTE stmt;
DEALLOCATE PREPARE stmt;
END ;;
DELIMITER ;
/*!50003 SET sql_mode            = @saved_sql_mode */ ;
/*!50003 SET character_set_client = @saved_cs_client */ ;
/*!50003 SET character_set_results = @saved_cs_results */ ;
/*!50003 SET collation_connection = @saved_col_connection */ ;
/*!50003 DROP PROCEDURE IF EXISTS `generar_horas` */ ;
/*!50003 SET @saved_cs_client      = @@character_set_client */ ;
/*!50003 SET @saved_cs_results     = @@character_set_results */ ;
/*!50003 SET @saved_col_connection = @@collation_connection */ ;
/*!50003 SET character_set_client  = utf8 */ ;
/*!50003 SET character_set_results = utf8 */ ;
/*!50003 SET collation_connection  = utf8_general_ci */ ;
/*!50003 SET @saved_sql_mode       = @@sql_mode */ ;
/*!50003 SET sql_mode              =
'STRICT_TRANS_TABLES,NO_AUTO_CREATE_USER,NO_ENGINE_SUBSTITUTION' */ ;
DELIMITER ;;
CREATE DEFINER='root'@'localhost' PROCEDURE `generar_horas`(param_horalIni TIME(5), param_horaFin
TIME(5), param_receso1Ini TIME(5), param_receso1Fin TIME(5), param_receso2Ini TIME(5), param_receso2Fin
TIME(5), usuario INT)
BEGIN
    DECLARE p_DuracionClase TIME(5);
    DECLARE horalni_TMP TIME(5);
    DECLARE horaFin_TMP TIME(5);
    DECLARE estado INT;
    SET horalni_TMP = param_horalni;

SET p_DuracionClase = (SELECT DuracionClase FROM
schollaryear
WHERE
ID_Estado = 1
ORDER BY FechaInicio DESC
LIMIT 1);

DELETE FROM Horas
WHERE
(Horalni BETWEEN param_horalni AND param_horaFin)
AND (HoraFin BETWEEN param_horalni AND param_horaFin);

WHILE (horalni_TMP <> param_horaFin) DO
    SET estado = 1;

    SELECT horalni_TMP, p_DuracionClase;
    SET horaFin_TMP = ADDTIME(horalni_TMP, p_DuracionClase);
SELECT horaFin_TMP;
    IF(horalni_TMP = param_receso1Ini) then
        SET estado = 2;

SET horaFin_TMP = param_receso1Fin;
    ELSEIF(horalni_TMP = param_receso2Ini) then
        SET estado = 2;

SET horaFin_TMP = param_receso2Fin;

```

```

END IF;

INSERT INTO `gaha`.`HORAS` (`Horalni`, `HoraFin`, `ID_Estado`, `FechaCreacion`,
`UsuarioCrea`)
VALUES(horalni_TMP, horaFin_TMP, estado, SYSDATE(), usuario);

IF(estado = 2) THEN
    SET horalni_TMP = horaFin_TMP;
ELSE
    SET horalni_TMP = ADDTIME(horalni_TMP, p_DuracionClase);
END IF;
END WHILE;
END ;;
DELIMITER ;
/*!50003 SET sql_mode            = @saved_sql_mode */ ;
/*!50003 SET character_set_client = @saved_cs_client */ ;
/*!50003 SET character_set_results = @saved_cs_results */ ;
/*!50003 SET collation_connection = @saved_col_connection */ ;
/*!50003 DROP PROCEDURE IF EXISTS `HCursos` */;
/*!50003 SET @saved_cs_client      = @@character_set_client */ ;
/*!50003 SET @saved_cs_results    = @@character_set_results */ ;
/*!50003 SET @saved_col_connection = @@collation_connection */ ;
/*!50003 SET character_set_client  = utf8 */ ;
/*!50003 SET character_set_results = utf8 */ ;
/*!50003 SET collation_connection  = utf8_general_ci */ ;
/*!50003 SET @saved_sql_mode      = @@sql_mode */ ;
/*!50003 SET sql_mode              =
'STRICT_TRANS_TABLES,NO_AUTO_CREATE_USER,NO_ENGINE_SUBSTITUTION' */ ;
DELIMITER ;;
CREATE DEFINER='root'@`localhost` PROCEDURE `HCursos`()
BEGIN
    SET @SQL = NULL;
SELECT
    GROUP_CONCAT(DISTINCT CONCAT('sum(case when r.id_curso = \'',
        dt,
        '\ then r.horassemana else 0 end) AS `',
        dt,
        '`'))
INTO @SQL FROM
    (SELECT
        d1.ID_curso AS dt
    FROM
        cursos d1
    ORDER BY d1.ID_curso) d;

SET @SQL
= CONCAT('SELECT r.id_materia AS id_materia, ', @SQL, '
from materiascursos r,
        cursos s
where r.id_curso = s.id_curso
group by r.id_materia
;');

PREPARE stmt FROM @SQL;
EXECUTE stmt;
DEALLOCATE PREPARE stmt;
END ;;
DELIMITER ;
/*!50003 SET sql_mode            = @saved_sql_mode */ ;
/*!50003 SET character_set_client = @saved_cs_client */ ;
/*!50003 SET character_set_results = @saved_cs_results */ ;

```

```

/*!50003 SET collation_connection = @saved_col_connection */;
/*!50003 DROP PROCEDURE IF EXISTS `HorarioProfesores` */;
/*!50003 SET @saved_cs_client = @@character_set_client */;
/*!50003 SET @saved_cs_results = @@character_set_results */;
/*!50003 SET @saved_col_connection = @@collation_connection */;
/*!50003 SET character_set_client = utf8 */;
/*!50003 SET character_set_results = utf8 */;
/*!50003 SET collation_connection = utf8_general_ci */;
/*!50003 SET @saved_sql_mode = @@sql_mode */;
/*!50003 SET sql_mode =
'STRICT_TRANS_TABLES,NO_AUTO_CREATE_USER,NO_ENGINE_SUBSTITUTION' */;
DELIMITER ;;
CREATE DEFINER='root'@'localhost' PROCEDURE `HorarioProfesores`()
BEGIN
    SET SESSION group_concat_max_len = 2056;
    SET @SQL = NULL;
    SELECT
        GROUP_CONCAT(DISTINCT CONCAT('sum(case when r.ID_Profesor = \'',
            dt,
            '\ then 1 else 0 end) AS `',
            dt,
            '`'))
INTO @SQL FROM
(SELECT
    d1.ID_Profesor AS dt
FROM
    profesores d1
ORDER BY d1.id_profesor) d;

    SET @SQL
        = CONCAT('SELECT r.id_dia AS dia , r.id_hora , @SQL, '
from dispprofesores r, profesores s
where r.id_profesor = s.id_profesor
group by r.id_dia, r.id_hora
;');

    PREPARE stmt FROM @SQL;
    EXECUTE stmt;
    DEALLOCATE PREPARE stmt;
END ;;
DELIMITER ;
/*!50003 SET sql_mode = @saved_sql_mode */;
/*!50003 SET character_set_client = @saved_cs_client */;
/*!50003 SET character_set_results = @saved_cs_results */;
/*!50003 SET collation_connection = @saved_col_connection */;
/*!50003 DROP PROCEDURE IF EXISTS `MateriaCursos` */;
/*!50003 SET @saved_cs_client = @@character_set_client */;
/*!50003 SET @saved_cs_results = @@character_set_results */;
/*!50003 SET @saved_col_connection = @@collation_connection */;
/*!50003 SET character_set_client = utf8 */;
/*!50003 SET character_set_results = utf8 */;
/*!50003 SET collation_connection = utf8_general_ci */;
/*!50003 SET @saved_sql_mode = @@sql_mode */;
/*!50003 SET sql_mode =
'STRICT_TRANS_TABLES,NO_AUTO_CREATE_USER,NO_ENGINE_SUBSTITUTION' */;
DELIMITER ;;
CREATE DEFINER='root'@'localhost' PROCEDURE `MateriaCursos`()
BEGIN
    SET @SQL = NULL;
    SELECT

```

```

GROUP_CONCAT(DISTINCT CONCAT('sum(case when r.id_curso = \'',
    dt,
    '\ then 1 else 0 end) AS ',
    dt,
    '''))
INTO @SQL FROM
(SELECT
    d1.ID_Curso AS dt
FROM
    cursos d1
ORDER BY d1.id_curso) d;

SET @SQL
= CONCAT('SELECT r.id_materia ', @SQL, '
from materiascursos r, cursos s
where r.id_curso = s.id_curso
group by r.id_materia
;');

PREPARE stmt FROM @SQL;
EXECUTE stmt;
DEALLOCATE PREPARE stmt;
END ;;
DELIMITER ;
/*!50003 SET sql_mode            = @saved_sql_mode */ ;
/*!50003 SET character_set_client = @saved_cs_client */ ;
/*!50003 SET character_set_results = @saved_cs_results */ ;
/*!50003 SET collation_connection = @saved_col_connection */ ;
/*!50003 DROP PROCEDURE IF EXISTS `MateriasAula` */;
/*!50003 SET @saved_cs_client      = @@character_set_client */ ;
/*!50003 SET @saved_cs_results    = @@character_set_results */ ;
/*!50003 SET @saved_col_connection = @@collation_connection */ ;
/*!50003 SET character_set_client  = utf8 */ ;
/*!50003 SET character_set_results = utf8 */ ;
/*!50003 SET collation_connection  = utf8_general_ci */ ;
/*!50003 SET @saved_sql_mode       = @@sql_mode */ ;
/*!50003 SET sql_mode              =
'STRICT_TRANS_TABLES,NO_AUTO_CREATE_USER,NO_ENGINE_SUBSTITUTION' */ ;
DELIMITER ;;
CREATE DEFINER='root'@'localhost' PROCEDURE `MateriasAula`()
BEGIN
    SET @SQL = NULL;
SELECT
    GROUP_CONCAT(DISTINCT CONCAT('sum(case when r.ID_TipoAula = \'',
        dt,
        '\ then 1 else 0 end) AS ',
        dt,
        '''))
INTO @SQL FROM
(SELECT
    d1.ID_TipoAula AS dt
FROM
    tipoaula d1
ORDER BY d1.ID_TipoAula) d;

SET @SQL
= CONCAT('SELECT r.id_materia AS id_materia, ', @SQL, '
from materias r,
            tipoaula s
where r.id_tipoaula = s.id_tipoaula
;');

```

```

        group by r.id_materia
        ;');

PREPARE stmt FROM @SQL;
EXECUTE stmt;
DEALLOCATE PREPARE stmt;
END ;;
DELIMITER ;
/*!50003 SET sql_mode            = @saved_sql_mode */ ;
/*!50003 SET character_set_client = @saved_cs_client */ ;
/*!50003 SET character_set_results = @saved_cs_results */ ;
/*!50003 SET collation_connection = @saved_col_connection */ ;
/*!50003 DROP PROCEDURE IF EXISTS `ProfesorMateria` */ ;
/*!50003 SET @saved_cs_client     = @@character_set_client */ ;
/*!50003 SET @saved_cs_results     = @@character_set_results */ ;
/*!50003 SET @saved_col_connection = @@collation_connection */ ;
/*!50003 SET character_set_client = utf8 */ ;
/*!50003 SET character_set_results = utf8 */ ;
/*!50003 SET collation_connection = utf8_general_ci */ ;
/*!50003 SET @saved_sql_mode       = @@sql_mode */ ;
/*!50003 SET sql_mode              =
'STRICT_TRANS_TABLES,NO_AUTO_CREATE_USER,NO_ENGINE_SUBSTITUTION' */ ;
DELIMITER ;;
CREATE DEFINER='root'@'localhost' PROCEDURE `ProfesorMateria`()
BEGIN
    SET SESSION group_concat_max_len = 2056;
    SET @SQL = NULL;
SELECT
    GROUP_CONCAT(DISTINCT CONCAT('sum(case when r.id_profesor = \',
        dt,
        '\ then 1 else 0 end) AS `',
        dt,
        '`'))
INTO @SQL FROM
    (SELECT
        d1.ID_Profesor AS dt
    FROM
        profesores d1
    ORDER BY d1.ID_Profesor) d;

SET @SQL
= CONCAT('SELECT r.id_materia AS id_materia, r.id_curso AS id_curso, ', @SQL, '
    from profesoresmaterias r,
        profesores s
    where r.id_profesor = s.id_profesor
    group by r.id_materia, r.id_curso
    ;');

PREPARE stmt FROM @SQL;
EXECUTE stmt;
DEALLOCATE PREPARE stmt;
END ;;
DELIMITER ;
/*!50003 SET sql_mode            = @saved_sql_mode */ ;
/*!50003 SET character_set_client = @saved_cs_client */ ;
/*!50003 SET character_set_results = @saved_cs_results */ ;
/*!50003 SET collation_connection = @saved_col_connection */ ;
/*!50003 DROP PROCEDURE IF EXISTS `registra_clave` */ ;
/*!50003 SET @saved_cs_client     = @@character_set_client */ ;
/*!50003 SET @saved_cs_results     = @@character_set_results */ ;
/*!50003 SET @saved_col_connection = @@collation_connection */ ;

```

```

/*!50003 SET character_set_client = utf8 */;
/*!50003 SET character_set_results = utf8 */;
/*!50003 SET collation_connection = utf8_general_ci */;
/*!50003 SET @saved_sql_mode = @@sql_mode */;
/*!50003 SET sql_mode =
'STRICT_TRANS_TABLES,NO_AUTO_CREATE_USER,NO_ENGINE_SUBSTITUTION' */;
DELIMITER ;;
CREATE DEFINER='root'@'localhost' PROCEDURE `registra_clave`(usuario varchar(15), pwd varchar(256))
BEGIN
UPDATE Usuarios
SET
    Password = SHA1(pwd)
WHERE
    ID_Usuario = (SELECT
        ID_Usuario
    FROM
        (SELECT
            a.ID_Usuario
        FROM
            Usuarios as a
        WHERE
            a.Usuario = usuario) AS UsuariosTMP);
END ;;
DELIMITER ;
/*!50003 SET sql_mode = @saved_sql_mode */;
/*!50003 SET character_set_client = @saved_cs_client */;
/*!50003 SET character_set_results = @saved_cs_results */;
/*!50003 SET collation_connection = @saved_col_connection */;
/*!40103 SET TIME_ZONE=@OLD_TIME_ZONE */;

/*!40101 SET SQL_MODE=@OLD_SQL_MODE */;
/*!40014 SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS */;
/*!40014 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS */;
/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
/*!40111 SET SQL_NOTES=@OLD_SQL_NOTES */;

```

4.3 Diccionario de Datos

Nombre de Objeto	arbolopciones	
Tipo de Objeto	tabla	
Descripción	Tabla del árbol para armar el menú de opciones	
Campo	Tipo de Dato	Comentarios
Hijo	int(11)	Opción hijo del menú
Padre	int(11)	Opción padre del menú
Profundidad	int(11)	Nivel de profundidad de la opción

Nombre de Objeto	aulas	
Tipo de Objeto	tabla	
Descripción	Tabla de aulas del plantel	
Campo	Tipo de Dato	Comentarios
Asientos	int(11)	Cantidad de asientos
Aula	varchar(10)	Nombre del aula
Comentarios	varchar(200)	
FechaCreacion	datetime	Fecha de creación del aula
FechaModificacion	datetime	Fecha de modificación del aula
ID_Aula	varchar(10)	Código del aula
ID_Estado	int(11)	Código interno del estado
ID_TipoAula	int(11)	Código interno del tipo de aula
UsuarioCrea	int(11)	Código interno del usuario de creación del aula
UsuarioModifica	int(11)	Código interno del usuario de modificación del aula

Nombre de Objeto	cursos	
Tipo de Objeto	tabla	
Descripción	Tabla de los cursos que imparte el plantel	
Campos	Tipo de Dato	Comentarios
Curso	varchar(20)	Nombre del curso
FechaCreacion	datetime	Fecha de creación del curso
FechaModificacion	datetime	Fecha de modificación del curso
ID_Curso	varchar(5)	Código del curso
ID_Estado	int(11)	Código interno del estado
Paralelos	int(11)	Cantidad de paralelos
UsuarioCrea	int(11)	Código interno del usuario de creación del curso
UsuarioModifica	int(11)	Código interno del usuario de modificación del curso

Nombre de Objeto	Días	
Tipo de Objeto	tabla	
Descripción	Tabla de los días de la semana	
Campos	Tipo de Dato	Comentarios
Dia	varchar(10)	Nombre del día
FechaCreacion	datetime	Fecha de creación del día
FechaModificacion	datetime	Fecha de modificación del día
ID_Dia	int(11)	Código interno del día
ID_Estado	int(11)	Código interno del estado
UsuarioCrea	int(11)	Código interno del usuario de creación del día
UsuarioModifica	int(11)	Código interno del usuario de modificación del día

Nombre de Objeto	dispprofesores	
Tipo de Objeto	tabla	
Descripción	Tabla de la disponibilidad de los profesores	
Campos	Tipo de Dato	Comentarios
FechaCreacion	datetime	Fecha de creación de la disponibilidad del profesor
FechaModificacion	datetime	Fecha de modificación de la disponibilidad del profesor
ID_Dia	int(11)	Código interno del día
ID_Estado	int(11)	Código interno del estado
ID_Hora	int(11)	Código interno de la hora
ID_Profesor	int(11)	Código interno del profesor
UsuarioCrea	int(11)	Código interno del usuario de creación de la disponibilidad del profesor
UsuarioModifica	int(11)	Código interno del usuario de modificación de la disponibilidad del profesor

Nombre de Objeto	estados	
Tipo de Objeto	tabla	
Descripción	Tabla de los estados del sistema	
Campos	Tipo de Dato	Comentarios
Descripcion	varchar(100)	Descripción del estado
Estado	varchar(50)	Nombre del estado
FechaCreacion	datetime	Fecha de creación del estado
FechaModificacion	datetime	Fecha de modificación del estado
ID_Estado	int(11)	Código interno del estado
UsuarioCrea	int(11)	Código interno del usuario de creación del estado
UsuarioModifica	int(11)	Código interno del usuario de modificación del estado

Nombre de Objeto	horarioscabecera	
Tipo de Objeto	tabla	
Descripción	Tabla de la cabecera de los horarios de clases	
Campos	Tipo de Dato	Comentarios
FechaCreacion	datetime	Fecha de creación de la cabecera del horario
FechaModificacion	datetime	Fecha de modificación de la cabecera del horario
ID_Estado	int(11)	Código interno del estado
ID_HorarioCab	int(11)	Código interno de la cabecera del horario
ID_SchollarYear	int(11)	Código interno del año escolar
UsuarioCrea	int(11)	Código interno del usuario de creación de la cabecera del horario
UsuarioModifica	int(11)	Código interno del usuario de modificación de la cabecera del horario

Nombre de Objeto	horarioscabeceratmp	
Tipo de Objeto	tabla	
descripción	Tabla temporal de la cabecera de los horarios de clases	
Campos	Tipo de Dato	Comentarios
FechaCreacion	datetime	Fecha de creación de la cabecera del horario
FechaModificacion	datetime	Fecha de modificación de la cabecera del horario
ID_Estado	int(11)	Código interno del estado
ID_HorarioCabTMP	int(11)	Código interno de la cabecera del horario
ID_SchollarYear	int(11)	Código interno del año escolar
UsuarioCrea	int(11)	Código interno del usuario de creación de la cabecera del horario
UsuarioModifica	int(11)	Código interno del usuario de modificación de la cabecera del horario

Nombre de Objeto	horariosdetalle	
Tipo de Objeto	tabla	
Descripción	Tabla del detalle de los horarios de clases	
Campos	Tipo de Dato	Comentarios
FechaCreacion	datetime	Fecha de creación del detalle del horario
FechaModificacion	datetime	Fecha de modificación del detalle del horario
ID_Aula	varchar(10)	Código del aula
ID_Curso	varchar(5)	Código interno del curso
ID_Dia	int(11)	Código interno del día
ID_Estado	int(11)	Código interno del estado
ID_Hora	int(11)	Código interno del periodo de clases
ID_HorarioCab	int(11)	Código interno de la cabecera del horario
ID_HorarioDet	int(11)	Código interno del detalle del horario
ID_Materia	varchar(20)	Código de la materia
ID_Profesor	int(11)	Código interno del profesor
Paralelo	varchar(5)	Paralelo
UsuarioCrea	int(11)	Código interno del usuario de creación del detalle del horario
UsuarioModifica	int(11)	Código interno del usuario de modificación del detalle del horario

Nombre de Objeto	horariosdetalletmp	
Tipo de Objeto	tabla	
Descripción	Tabla temporal del detalle de los horarios de clases	
Campos	Tipo de Dato	Comentarios
FechaCreacion	datetime	Fecha de creación del detalle del horario
FechaModificacion	datetime	Fecha de modificación del detalle del horario
ID_Aula	varchar(10)	Código del aula
ID_Curso	varchar(5)	Código interno del curso
ID_Dia	varchar(5)	Código interno del día
ID_Estado	int(11)	Código interno del estado
ID_Hora	int(11)	Código interno del periodo de clases
ID_HorarioCabTMP	int(11)	Código interno de la cabecera del horario
ID_HorariosDetTMP	int(11)	Código interno del detalle del horario
ID_Materia	varchar(20)	Código de la materia
ID_Profesor	int(11)	Código interno del profesor
Paralelo	varchar(5)	Paralelo
UsuarioCrea	int(11)	Código interno del usuario de creación del detalle del horario
UsuarioModifica	int(11)	Código interno del usuario de modificación del detalle del horario

Nombre de Objeto	horas	
Tipo de Objeto	tabla	
Descripción	Tabla de los periodos de clases	
Campos	Tipo de Dato	Comentarios
Comentarios	varchar(200)	
FechaCreacion	datetime	Fecha de creación del periodo de clases
FechaModificacion	datetime	Fecha de modificación del periodo de clases
HoraFin	time(5)	Hora final del periodo de clases
Horalni	time(5)	Hora inicial del periodo de clases
ID_Estado	int(11)	Código interno del estado
ID_Hora	int(11)	Código interno del periodo de clases
UsuarioCrea	int(11)	Código interno del usuario de creación del periodo de clases
UsuarioModifica	int(11)	Código interno del usuario de modificación del periodo de clases

Nombre de Objeto	materias	
Tipo de Objeto	tabla	
Descripción	Tabla de las materias que imparte el plantel	
Campos	Tipo de Dato	Comentarios
FechaCreacion	datetime	Fecha de creación de la materia
FechaModificacion	datetime	Fecha de modificación de la materia
ID_Estado	int(11)	Código interno del estado
ID_Materia	varchar(20)	Código de materia
ID_TipoAula	int(11)	Código interno del tipo de aula
Materia	varchar(100)	Nombre de la materia
UsuarioCrea	int(11)	Código interno del usuario de creación de la materia
UsuarioModifica	int(11)	Código interno del usuario de modificación de la materia

Nombre de Objeto	materiascursos	
Tipo de Objeto	tabla	
Descripción	Tabla de la relación de las materias por cursos	
Campos	Tipo de Dato	Comentarios
FechaCreacion	datetime	Fecha de creación de la materia por curso
FechaModificacion	datetime	Fecha de modificación de la materia por curso
HorasSemana	int(11)	Horas a la semana de la materia por curso
ID_Curso	varchar(5)	Código del curso
ID_Estado	int(11)	Código interno del estado
ID_Materia	varchar(20)	Código de la materia
UsuarioCrea	int(11)	Código interno del usuario de creación de la materia por curso
UsuarioModifica	int(11)	Código interno del usuario de modificación de la mateira por curso

Nombre de Objeto	opciones	
Tipo de Objeto	tabla	
Descripción	Tabla de opciones del sistema	
Campos	Tipo de Dato	Comentarios
FechaCreacion	datetime	Fecha de creación de la opción
FechaModificacion	datetime	Fecha de modificación de la opción
ID_Estado	int(11)	Código interno del estado
ID_Opcion	int(11)	Código interno de la opción
ID_Permiso	int(11)	Código interno del permiso de la opción
Imagen	varchar(150)	Imagen de la opción
Opcion	varchar(100)	Nombre de la opción
OpcionContenedora	int(11)	Código interno de la opción contenedora
Orden	tinytext	Orden de la opción
TituloMenu	varchar(50)	Título de menú de la opción
URL	varchar(150)	Dirección url de la opción
UsuarioCrea	int(11)	Código interno del usuario de creación de la opción
UsuarioModifica	int(11)	Código interno del usuario de modificación de la opción

Nombre de Objeto	perfilesusuario	
Tipo de Objeto	tabla	
Descripción	Tabla de los perfiles de usuarios	
Etiquetas de fila	Tipo de Dato	Comentarios
Comentarios	varchar(200)	
FechaCreacion	datetime	Fecha de creación del perfil de usuario
FechaModificacion	datetime	Fecha de modificación del perfil de usuario
ID_Estado	int(11)	Código interno del estado
ID_PerfilUsuario	int(11)	Código interno del perfil de usuario
Perfil	varchar(60)	Nombre del perfil de usuario
UsuarioCrea	int(11)	Código interno del usuario de creación del perfil de usuario
UsuarioModifica	int(11)	Código interno del usuario de modificación del perfil de usuario

Nombre de Objeto	permisos	
Tipo de Objeto	tabla	
Descripción	Tabla de los permisos del sistema	
Etiquetas de fila	Tipo de Dato	Comentarios
FechaCreacion	datetime	Fecha de creación del permiso
FechaModificacion	datetime	Fecha de modificación del permiso
ID_Estado	int(11)	Código interno del estado
ID_Permiso	int(11)	Código interno del permiso
Permiso	varchar(60)	Nombre del permiso
UsuarioCrea	int(11)	Código interno del usuario de creación del permiso
UsuarioModifica	int(11)	Código interno del usuario de modificación del permiso

Nombre de Objeto	permisosperfiles	
Tipo de Objeto	tabla	
Descripción	Tabla de la relación de los permisos asignados a un perfil	
Etiquetas de fila	Tipo de Dato	Comentarios
FechaCreacion	datetime	Fecha de creación de los permisos del perfil
FechaModificacion	datetime	Fecha de modificación de los permisos del perfil
ID_Estado	int(11)	Código interno del estado
ID_Perfil	int(11)	Código interno del perfil de usuario
ID_Permission	int(11)	Código interno del permiso
UsuarioCrea	int(11)	Código interno del usuario de creación de los permisos del perfil
UsuarioModifica	int(11)	Código interno del usuario de modificación de los permisos del perfil

Nombre de Objeto	profesores	
Tipo de Objeto	tabla	
Descripción	Tabla de los profesores del panel	
Etiquetas de fila	Tipo de Dato	Comentarios
FechaCreacion	datetime	Fecha de creación del profesor
FechaModificacion	datetime	Fecha de modificación del profesor
ID_Estado	int(11)	Código interno del estado
ID_Profesor	int(11)	Código interno del profesor
Profesor	varchar(100)	Nombre del profesor
Título	varchar(100)	Título del profesor
UsuarioCrea	int(11)	Código interno del usuario de creación del profesor
UsuarioModifica	int(11)	Código interno del usuario de modificación del profesor

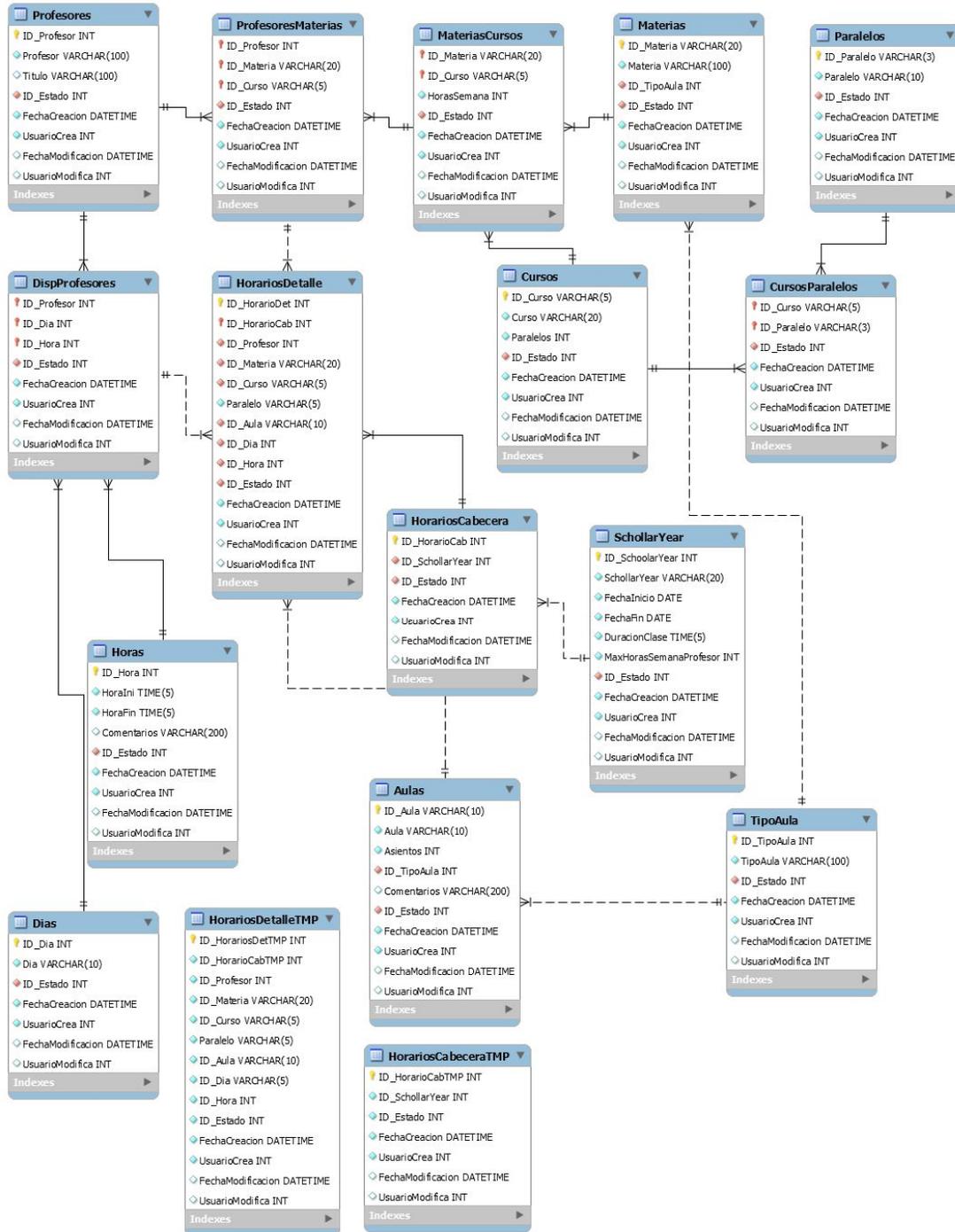
Nombre de Objeto	profesoresmaterias	
Tipo de Objeto	tabla	
Descripción	Tabla de la relación de las materias que puede dar el profesor	
Etiquetas de fila	Tipo de Dato	Comentarios
FechaCreacion	datetime	Fecha de creación de las materias por curso del profesor
FechaModificacion	datetime	Fecha de modificación de las materias por curso del profesor
ID_Curso	varchar(5)	Código del curso
ID_Estado	int(11)	Código interno del estado
ID_Materia	varchar(20)	Código de la materia
ID_Profesor	int(11)	Código interno del profesor
UsuarioCrea	int(11)	Código interno del usuario de creación de las materias por curso del profesor
UsuarioModifica	int(11)	Código interno del usuario de modificación de las materias por curso del profesor

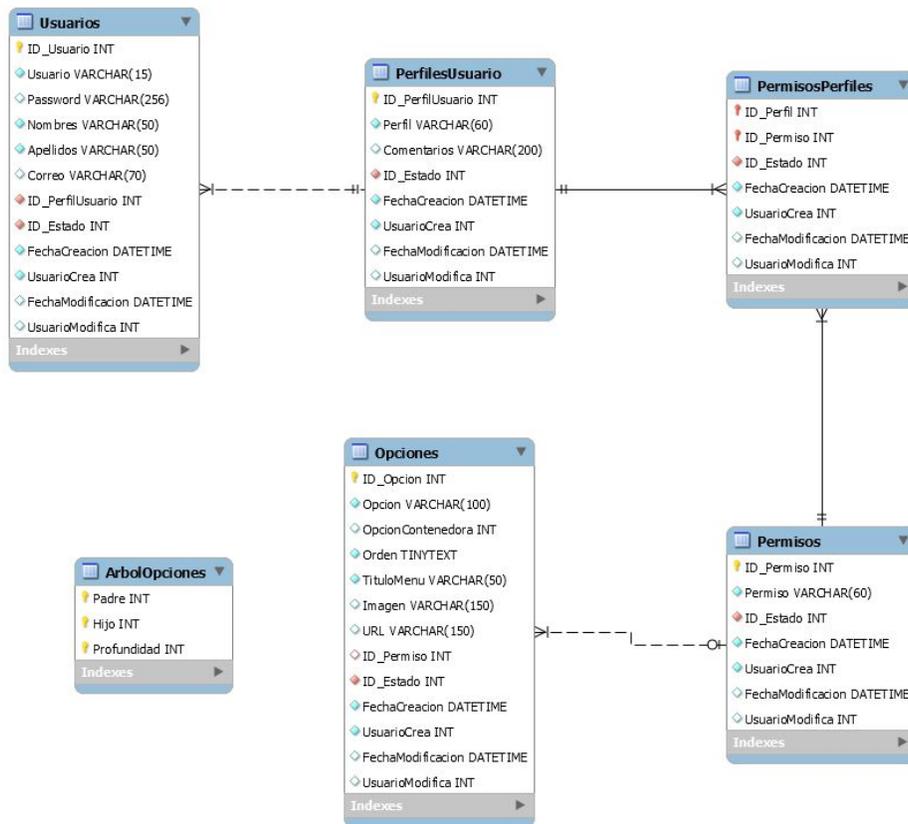
Nombre de Objeto	schollaryear	
Tipo de Objeto	tabla	
Descripción	Tabla de los años escolares por aperturar	
Etiquetas de fila	Tipo de Dato	Comentarios
DuracionClase	time(5)	Duración de las clases
FechaCreacion	datetime	Fecha de creación del año lectivo
FechaFin	date	Fecha de fin del año lectivo
FechaInicio	date	Fecha de inicio del año lectivo
FechaModificacion	datetime	Fecha de modificación del año lectivo
ID_Estado	int(11)	Código interno del estado
ID_SchoolarYear	int(11)	Código interno del año lectivo
MaxHorasSemanaProfesor	int(11)	Máximo de horas a la semana del profesor
SchollarYear	varchar(20)	Año lectivo
UsuarioCrea	int(11)	Código interno del usuario de creación del año lectivo
UsuarioModifica	int(11)	Código interno del usuario de modificación del año lectivo

Nombre de Objeto	tipoaula	
Tipo de Objeto	tabla	
Descripción	Tabla de los tipos de aulas existentes en el panel	
Etiquetas de fila	Tipo de Dato	Comentarios
FechaCreacion	datetime	Fecha de creación del tipo de aula
FechaModificacion	datetime	Fecha de modificación del tipo de aula
ID_Estado	int(11)	Código interno del estado
ID_TipoAula	int(11)	Código interno del tipo de aula
TipoAula	varchar(100)	Nombre del tipo de aula
UsuarioCrea	int(11)	Código interno del usuario de creación del tipo de aula
UsuarioModifica	int(11)	Código interno del usuario de modificación del tipo de aula

Nombre de Objeto	usuarios	
Tipo de Objeto	tabla	
Descripción	Tabla de los usuarios del sistema	
Etiquetas de fila	Tipo de Dato	Comentarios
Apellidos	varchar(50)	Apellidos
Correo	varchar(70)	Correo electrónico
FechaCreacion	datetime	Fecha de creación del usuario
FechaModificacion	datetime	Fecha de modificación del usuario
ID_Estado	int(11)	Código interno del estado
ID_PerfilUsuario	int(11)	Código interno del perfil de usuario
ID_Usuario	int(11)	Código interno del usuario
Nombres	varchar(50)	Nombres
Password	varchar(256)	Contraseña
Usuario	varchar(15)	Nombre de usuario
UsuarioCrea	int(11)	Código interno del usuario de creación del usuario
UsuarioModifica	int(11)	Código interno del usuario de modificación del usuario

4.4 Modelo Entidad Relación







DECLARACIÓN Y AUTORIZACIÓN

Nosotros, **Borja Gutiérrez, Petita María**, con C.C. #1312213380 y **Triviño Calderón, Jean Carlos**, con C.C. #0925467250, autores del trabajo de titulación: **Desarrollo e implementación de un sistema web gestor de horarios académicos automáticos para la Unidad Educativa Federico González Suarez del cantón Durán**, previo a la obtención del título de **INGENIERO EN SISTEMAS COMPUTACIONALES** en la Universidad Católica de Santiago de Guayaquil.

1.- Declaramos tener pleno conocimiento de la obligación que tienen las instituciones de educación superior, de conformidad con el Artículo 144 de la Ley Orgánica de Educación Superior, de entregar a la SENESCYT en formato digital una copia del referido trabajo de titulación para que sea integrado al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.

2.- Autorizamos a la SENESCYT a tener una copia del referido trabajo de titulación, con el propósito de generar un repositorio que democratice la información, respetando las políticas de propiedad intelectual vigentes.

Guayaquil, 23 de Septiembre de 2016

f. 

Nombre: **Borja Gutiérrez, Petita María**

C.C: **1312213380**

f. 

Nombre: **Triviño Calderón, Jean Carlos**

C.C: **0925467250**



REPOSITORIO NACIONAL EN CIENCIA Y TECNOLOGÍA

FICHA DE REGISTRO DE TESIS/TRABAJO DE TITULACIÓN

TÍTULO Y SUBTÍTULO:	Desarrollo e implementación de un sistema web gestor de horarios académicos automáticos para la Unidad Educativa Federico González Suárez del cantón Durán		
AUTOR(ES)	Borja Gutiérrez, Petita María; Triviño Calderón, Jean Carlos		
REVISOR(ES)/TUTOR(ES)	Sánchez Calle, Roberto Eduardo		
INSTITUCIÓN:	Universidad Católica de Santiago de Guayaquil		
FACULTAD:	Facultad de Ingeniería		
CARRERA:	Escuela de Ingeniería en Sistemas		
TITULO OBTENIDO:	Ingeniero en Sistemas Computacionales		
FECHA DE PUBLICACIÓN:		No. DE PÁGINAS:	160
ÁREAS TEMÁTICAS:	Sistemas de Información, Desarrollo de Sistemas		
PALABRAS CLAVES/ KEYWORDS:	HORARIOS; COLEGIO; ALGORITMO; DESARROLLO; SISTEMA WEB; CERTIFICADOS DIGITALES; FRAMEWORK; SERVIDOR WEB		
RESUMEN/ABSTRACT	<p>La solución propuesta en el presente proyecto de titulación, es el desarrollo e implementación del sistema web de generación automática de horarios académicos para la Unidad Educativa Federico González Suárez del cantón Duran, que nace a partir de la problemática del desgaste de tiempo de los recursos asignados durante la elaboración manual de los horarios de clases del año lectivo por aperturar, siendo uno de los objetivos del sistema optimizar el tiempo de elaboración de los horarios de clase a través de la automatización del mismo, considerando los cursos y paralelos aperturados, materias vigentes, tipo de aula que requiere la materia, materias por curso, materias que puede impartir el profesor, la disponibilidad del profesor y los periodos de clases de acuerdo a la duración de clases establecida sin considerar los recesos establecidos por el plantel, teniendo acceso a la visualización de la información del sistema de acuerdo al perfil del usuario correspondiente y desde cualquier dispositivo conectado a internet.</p>		
ADJUNTO PDF:	<input checked="" type="checkbox"/> SI	<input type="checkbox"/> NO	
CONTACTO CON AUTOR/ES:	Teléfono: 0989786856/ 0992159050	E-mail: peti.borja@hotmail.com / jeanca93@gmail.com	
CONTACTO CON LA INSTITUCIÓN: COORDINADOR DEL PROCESO DE UTE	Nombre: Valencia Macias, Lorgia del Pilar		
	Teléfono: +593-4-2206950 ext 1020		
	E-mail: lorgia.valencia@cu.ucsg.edu.ec		
SECCIÓN PARA USO DE BIBLIOTECA			
Nº. DE REGISTRO (en base a datos):			
Nº. DE CLASIFICACIÓN:			
DIRECCIÓN URL (tesis en la web):			