



# DASC INSTITUTO TECNOLÓGICO UNIVERSITARIO

## PROYECTO FINAL

DESARROLLO DE UN SITIO WEB  
DE CONTROL ESCOLAR PARA EL INSTITUTO  
CUAUHTÉMOC

HÉCTOR COYOTL JUAREZ – ID19191

PROFESOR: JOSE GERARDO GOMEZ VELAZQUEZ

# INDICE

<i>INTRODUCCIÓN</i> .....	1
Presentación .....	1
Contexto .....	1
Estructura del documento.....	1
<i>ESPECIFICACIÓN DE REQUISITOS</i> .....	2
Propósito .....	2
Definiciones, siglas y abreviaturas .....	2
Visión global .....	3
Descripción general.....	3
Funciones del producto .....	3
<i>RESTRICCIONES</i> .....	5
<i>ANÁLISIS</i> .....	6
Introducción .....	6
Diagrama de casos de uso .....	6
<i>DISEÑO</i> .....	10
Capa de persistencia o datos.....	10
<i>TECNOLOGÍAS UTILIZADAS</i> .....	15
<i>CODIGO DEL PROYECTO</i> .....	17

# INTRODUCCIÓN

## Presentación

Este documento describe el trabajo realizado en el proyecto final de carrera de ingeniería de software. El proyecto consiste en el desarrollo del sitio web de control escolar del Instituto Cuauhtémoc, situado en la localidad de San Ramón, 72490 Puebla, Pue. El propósito de este sitio web es brindar información completa sobre la escuela y los servicios disponibles, así como proporcionar mejores herramientas para los estudiantes y maestros del Centro Educativo.

## Contexto

Para el desarrollo de la aplicación realicé diversas reuniones con la directora del colegio y se validó como realiza sus procesos, para después poder informatizarlas adecuadamente. La mayoría de las decisiones de estética y funcionales de la aplicación fueron tomadas junto con la directora, que además me proporcionó toda la documentación y reglas del colegio.

Una vez realizadas las reuniones con la directora y estudiada la situación, distribuí el trabajo en 3 partes: Modulo de administrador, Modulo de Usuario, Modulo de profesores.

## Estructura del documento

Este documento se divide básicamente en varios capítulos correspondientes a diferentes fases del proceso de desarrollo del proyecto. Estos niveles:

**Especificación de requisitos:** La visión inicial del proyecto se redactó de manera global y describió los requisitos que debían cumplirse. El propósito de esta fase es registrar el acuerdo entre el desarrollador y el cliente con respecto a la funcionalidad del proyecto.

**Análisis:** El modelado conceptual de soluciones futuras se basó en diagramas. Los modelos nos ayudaran a visualizar el sistema. Esta fase especifica qué debe hacer la aplicación, pero no cómo.

**Diseño:** Utilizando los elementos y modelos obtenidos durante el análisis, los hemos transformado en mecanismos que se pueden utilizar en un entorno web con propiedades y condiciones para crear este tipo de entornos. Todos los niveles de la aplicación están diseñados. No se requiere código durante las etapas de análisis y diseño. Un buen análisis y un buen diseño son las mejores formas de crear software de calidad.

## ESPECIFICACIÓN DE REQUISITOS

### Propósito

El propósito de la especificación de requisitos es definir cuáles son los requerimientos que debe tener la aplicación que se va a desarrollar y describir la funcionalidad del usuario a lo largo de ella.

### Definiciones, siglas y abreviaturas

**Sitio web:** Conjunto de archivos electrónicos y páginas web referentes a un tema en particular que incluye una página inicial de bienvenida, con un nombre de dominio y dirección en Internet específicos.

**Interfaz:** Parte del programa informático que permite el flujo de información entre varias aplicaciones o entre el propio programa y el usuario.

**Navegador:** Permite al usuario recuperar y visualizar páginas web a través de Internet.

**Servidor web:** Se trata de un programa que implementa el protocolo HTTP (HyperText Transfer Protocol). Este protocolo está diseñado para transferir lo que llamamos hipertextos, páginas web o páginas HTML: textos complejos con enlaces, figuras, formularios, botones y objetos incrustados como animaciones o reproductores de música.

**GUI:** Es la abreviatura de Graphic User Interface o interfaz gráfica de usuario.

Es cualquier artefacto gráfico que permite a los usuarios interactuar con una aplicación usando iconos, botones, indicadores visuales, etc.... en contraste con las interfaces más tradicionales basadas en texto, o las más avanzadas actualmente basadas en la voz o en la interacción mediante movimientos.

**IDE:** Un Entorno Integrado de Desarrollo o IDE (del inglés Integrated Development Environment) es una aplicación para desarrollar aplicaciones y que va mucho más allá de lo que ofrece un simple editor.

**Cliente-servidor:** Es una estructura de aplicación distribuida que divide tareas o cargas de trabajo entre los proveedores de un recurso o servicio, denominados servidores, y los solicitantes del servicio, denominados clientes.

**HTTP:** El Protocolo de transferencia de hipertexto es el protocolo de comunicación que permite las transferencias de información a través de archivos en la World Wide Web.

## **Visión global**

A continuación, se realizará la descripción general del sistema desarrollado con sus funciones, características del usuario, restricciones, supuestos y dependencias. También se expondrá una especificación detallada de los requisitos detectados.

## **Descripción general**

### **Perspectiva del producto**

La aplicación desarrollada está destinada a proporcionar información específica para cada tipo de usuario registrado. Se puede acceder a esta aplicación desde los siguientes sistemas operativos (Windows, Linux, Android y Mac) siempre conectado a Internet a través de un navegador web se recomienda (Microsoft Edge y Google Chrome)

## **Funciones del producto**

A continuación, se muestran las funciones que conforman la aplicación, según el tipo de usuario que se encuentre logueado.

### **Usuario alumno/padre**

Consultar horario: Se muestra un listado de las asignaturas en las que el alumno tomara a lo largo de su curso (día de la semana, hora de inicio y hora de fin).

Consultar pagos: Se muestra un listado de los pagos realizados a lo largo de su estadía.

Consultar clases y grupo: Se muestra un listado de las materias clases del día y horarios.

Listar profesores: Se listarán los profesores que imparten clase al alumno (nombre, apellidos y nombre de la asignatura que le imparten).

El alumno podrá realizar las actividades asignadas por el profesor.

El alumno podrá validar las calificaciones de las actividades realizadas de todas sus materias.

### **Usuario profesor**

Listar alumnos: Se muestra el listado de las asignaturas que imparte el profesor cuando selecciona una de ellas se muestran los alumnos que se encuentra en esa clase.

Listar grupos: Se muestra el listado de los grupos asignados.

Mostrará los correspondientes días y horarios en los cuales el profesor impartirá clase.

El profesor podrá crear actividades de su materia para que los alumnos la realicen, tendrá una fecha de inicio y fin en este apartado se podrá habilitar para adjuntar archivos.

Estas actividades se podrán editar las fechas de entrega y eliminar en el caso de que se requieran.

El profesor calificará las actividades y podrá agregar una retroalimentación a la actividad si lo requiere.

### **Usuario administrador**

Registrar alumno: Se podrá dar de alta alumno.

Consultar alumno: Se podrá consultar de manera rápida y precisa los datos del alumno.

Gestionar alumno: Se podrá modificar los datos de un alumno.

Registrar profesor: Se podrá dar de alta un profesor.

Consultar profesor: Se podrá consultar de manera rápida y precisa los datos del profesor.

Gestionar profesor: Se podrá modificar los datos de un profesor.

Registrar materias: Se podrá dar de alta unas materias.

Consultar materias: Se podrá consultar de manera rápida y precisa los datos de las materias.

Gestionar materias: Se podrá modificar los datos de unas materias.

Registrar horarios: Se podrá dar de alta horarios.

Consultar horarios: Se podrá consultar de manera rápida y precisa los datos de los horarios.

Gestionar horarios: Se podrá modificar los datos de los horarios.

Registrar grados: Se podrá dar de alta grados.

Consultar grados: Se podrá consultar de manera rápida y precisa los datos de los grados.

Gestionar grados: Se podrá modificar los datos de los grados.

Registrar pagos: Se podrá registrar los pagos de los alumnos.

Consultar pagos: Se podrá consultar de manera rápida y precisa los pagos de los alumnos.

Gestionar pagos: Se podrá modificar los datos de los pagos.

Registrar aulas: Se podrá registrar nuevas aulas.

Consultar aulas: Se podrá consultar de manera rápida y precisa todas las aulas de la escuela.

Gestionar aulas: Se podrá modificar los datos de las aulas.

Registrar periodos: Se podrá registrar los periodos correspondientes.

Consultar periodos: Se podrá consultar de manera rápida y precisa los diferentes periodos y su estatus.

Gestionar periodos: Se podrá modificar los datos de cada uno de los periodos.

Consultar el detalle de los alumnos asignados a cada profesor y su correspondiente materia, grado y aula en la cual se encuentre.

Se podrá gestionar la asignación de las materias a los profesores y a su vez asignarle alumnos.

Se requiere que solo el administrador realice el alta de los diferentes usuarios y le asigna el rol correspondiente.

Consultar total de alumnos: Se podrá consultar de manera rápida y precisa toda la cantidad de alumnos en la escuela.

Consultar total de profesores: Se podrá consultar de manera rápida y precisa toda la cantidad de profesores en la escuela.

Consultar pagos del día: Se podrá consultar de manera rápida el total de pagos realizado en el día.

Consultar pagos por concepto: Se podrá consultar de manera rápida en una grafica los diferentes pagos realizados en el mes estos estarán agrupados por concepto.

## **RESTRICCIONES**

Al tratarse de una aplicación web, se requiere un ordenador con un navegador convencional y una conexión a Internet básica.

Este proyecto se estará mostrando de manera local, no cuenta con un dominio público.

Se tiene que asignar un servidor para el alojamiento del sitio web en caso de no contar con ello se tendrá que adquirir con los diferentes proveedores en el mercado actual.

Supuestos y dependencias

La aplicación desarrollada trabaja al margen de cualquier hardware o software ofreciendo así un soporte multiplataforma. La única dependencia importante que podemos encontrar está relacionada con el servidor web donde se encuentre alojado nuestro portal, el cual ha de ser capaz de soportar PHP Versión 7.4.30, MySQL Workbench 8.0.

## **ANÁLISIS**

### **Introducción**

Para realizar el análisis de esta aplicación web se ha optado por seguir las recomendaciones definidas por UML (Unified Modeling Language o Lenguaje Unificado de Modelado en castellano). Este modelo dispone de multitud de diagramas que ayudan a comprender la complejidad del futuro sistema, permitiéndonos plasmar en un lenguaje estándar aquellas funcionalidades, requisitos y demás características que hemos detectado en el sistema.

Aunque UML define una gran cantidad de diagramas para representar los distintos aspectos del desarrollo de la aplicación, en nuestro caso nos hemos centrado en diagrama de casos de usos.

### **Diagrama de casos de uso**

Los diagramas de casos de uso nos permiten diferenciar los actores que interactúan con nuestra aplicación, las relaciones entre ellos y las acciones que puede realizar cada uno dentro del sistema. Este tipo de diagramas son fácilmente comprensibles tanto por clientes como por usuarios, representan los requisitos funcionales del sistema y se utilizan como base para un desarrollo iterativo e incremental. Los diagramas de casos de uso tienen tres elementos:

**Actores:** Son los usuarios del sistema (figura 1). Un actor puede ser una persona, un conjunto de personas, un sistema hardware o un sistema software. Los actores representan un rol, que puede desempeñar alguien que necesita intercambiar información con el sistema.

**Casos de uso:** Un caso de uso describe una forma concreta de utilizar parte de la funcionalidad de un sistema. La colección de todos los casos de uso describe toda la funcionalidad del sistema.

**Comunicación entre actores y casos de uso:** Cada actor ejecuta un número específico de casos de uso en la aplicación. Por eso decimos que hay comunicación entre actores y casos de uso.



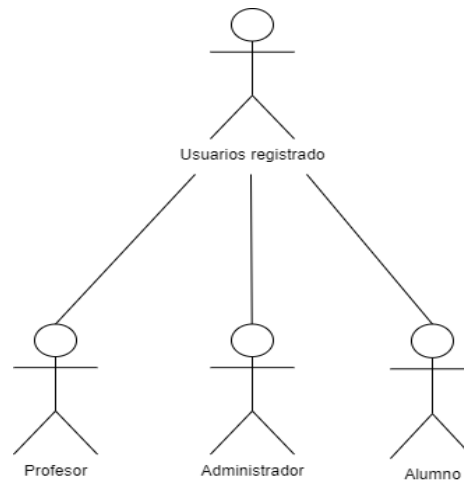


Figura 1. Actores

Los usuarios registrados son los alumnos y profesores de la institución y el administrador, que acceden a la aplicación mediante un nombre de usuario y contraseña.

### Casos de uso del alumno

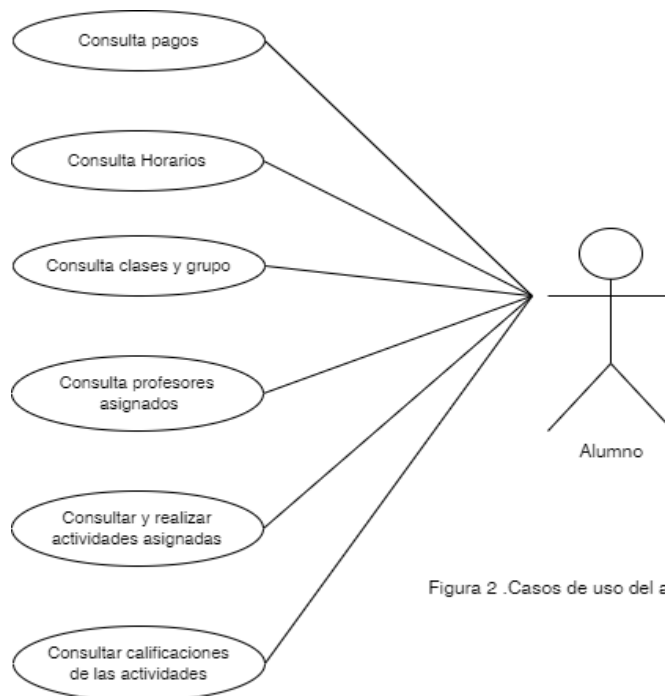
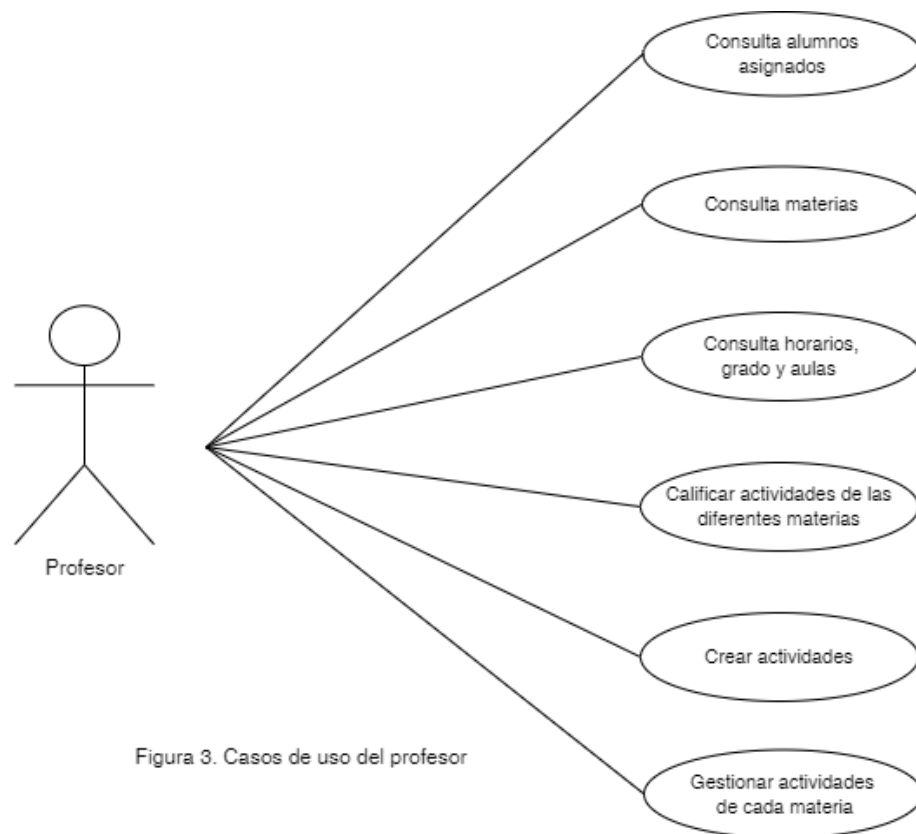


Figura 2 .Casos de uso del alumno.

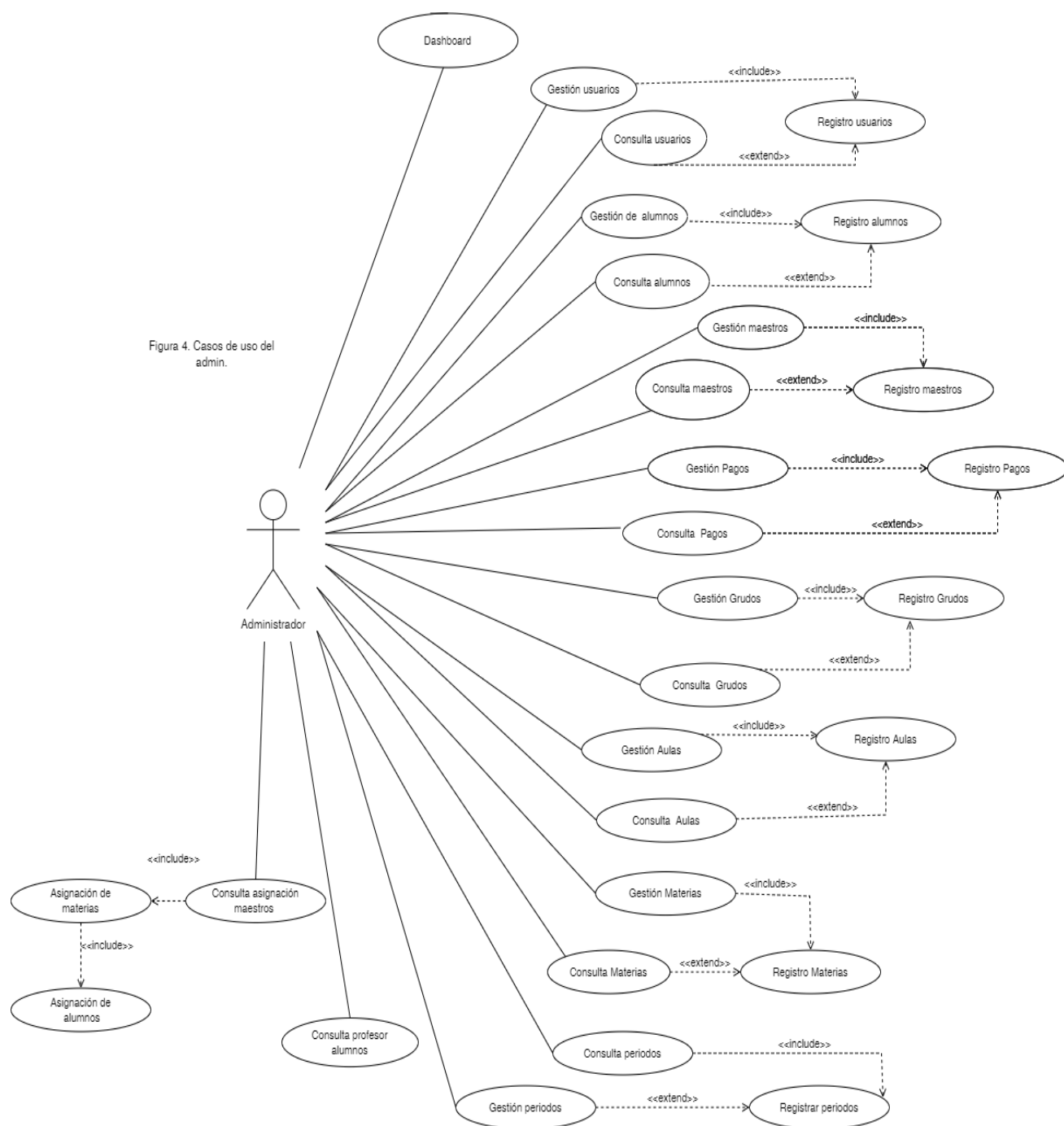
Este tipo de usuario representa a un alumno del colegio. Podrá realizar las acciones Login los módulos que tiene acceso son los siguientes Pagos, Materias y Actividades.

## Casos de uso del profesor



Este tipo de usuario representa a un profesor que imparte clases en el colegio. Podrá realizar las acciones de consultar alumnos asignados, materias, horarios, grados y aulas. Cada profesor podrá crear actividades de su materia para realizar en la escuela o de tarea estas mismas las podrá calificar una vez que el alumno las concluya.

## Casos de uso del administrador



Este tipo de usuario representa al administrador de la aplicación. Podrá realizar las acciones de consulta, gestión, registro de todos los módulos.

## **DISEÑO**

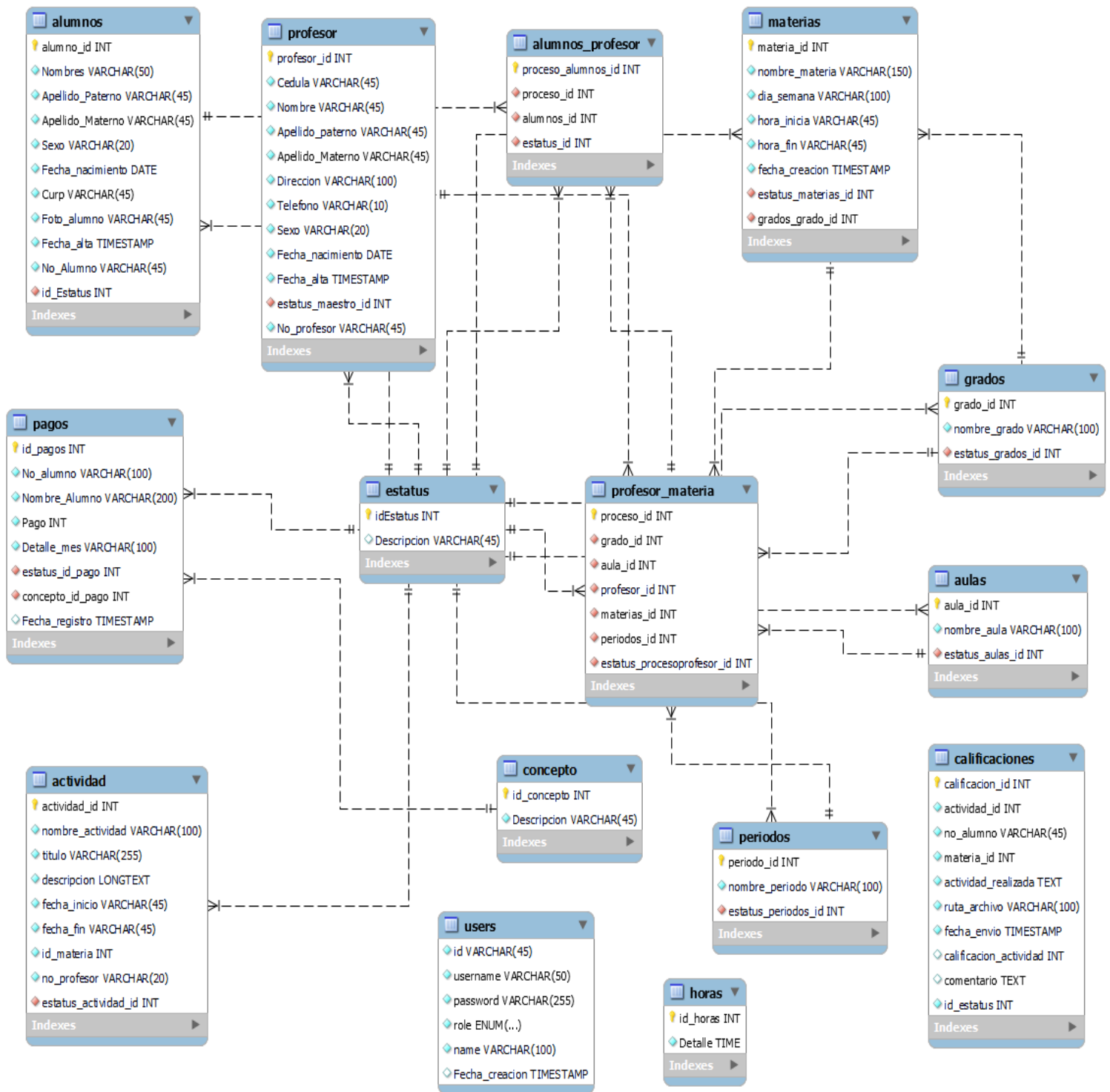
### **Capa de persistencia o datos**

La capa de persistencia se puede definir como el conjunto de componentes software que proporcionan una serie de servicios que permiten a los objetos del dominio interactuar con su repositorio permanente asociado.

En el proyecto la capa de persistencia se corresponde con la base de datos de la aplicación y las distintas Tablas: actividad, alumnos, alumnos\_profesor, aulas, calificaciones, concepto, estatus, grados, horas, materias, pagos, periodos, profesor, profesor\_materia y users.

Vistas: vw\_detalle\_actividad, vw\_detalle\_actividad, vw\_detalle\_alumnosasignados, vw\_detalle\_calificacion, vw\_detalle\_maestros, vw\_detalle\_materias, vw\_detalle\_pagos, vw\_detalle\_profesormateria, vw\_nombres, vw\_nombres\_maestros.

A continuación, se muestra el diagrama entidad-relación utilizado para la implementación de la base de datos de nuestro proyecto.



vw\_detalle\_alumnos    vw\_detalle\_maestros    vw\_nombres    vw\_detalle\_calificacion    vw\_detalle\_materias  
 vw\_detalle\_profesormateria    vw\_nombres\_maestros    vw\_detalle\_alumnosasignados    vw\_detalle\_actividad    vw\_detalle\_pagos

## Descripción de los campos de cada una de las tablas

TABLA : actividad

NOMBRE DEL CAMPO	TIPO DE CAMPO	DESCRIPCION
actividad_id	int AI PK	Identificador único para cada actividad introducida. Clave primaria.
nombre_actividad	varchar(100)	Nombre de la actividad no mayor a 100 caracteres.
titulo	varchar(255)	Título de cada actividad no mayor a 255 caracteres.
descripcion	text	Descripción de la actividad.
fecha_inicio	varchar(45)	Fecha de inicio de la actividad.
fecha_fin	varchar(45)	Fecha fin de la cual la actividad concluye.
id_materia	int	Identificador único de cada materia.
no_profesor	varchar(20)	No profesor del cual esta logueado.
estatus_actividad_id	int	Identificador único del estatus.

TABLA : alumnos

NOMBRE DEL CAMPO	TIPO DE CAMPO	DESCRIPCION
alumno_id	int AI PK	Identificador único para cada alumno registrado. Clave primaria.
Nombres	varchar(50)	Nombre del alumnos.
Apellido_Paterno	varchar(45)	Apellido pateno.
Apellido_Materno	varchar(45)	Apellido materno.
Sexo	varchar(20)	Sexo : Femenino / Masculino
Fecha_nacimiento	date	Fecha de nacimiento dia/mes/año.
Curp	varchar(45)	Curp del alumno a registrar.
Foto_alumno	varchar(45)	En este campo solo se inserta la ruta de la imagen.
Fecha_alta	timestamp	Este campo se agrega automatica mente una vez que se realiza el inset.
No_Alumno	varchar(45)	No alumno en el formulario se lo asigna automatica mente.
id_Estatus	int	Identificador único del estatus.

TABLA : alumnos\_profesor

NOMBRE DEL CAMPO	TIPO DE CAMPO	DESCRIPCION
proceso_alumnos_id	int AI PK	Identificador único para cada registro de alumno profesor . Clave primaria.
proceso_id	int	Identificador único del proceso tipo de dato numeros enteros.
alumnos_id	int	Identificador único del alumno tipo de dato numeros enteros.
estatus_id	int	Identificador único del estatus tipo de dato numeros enteros.

TABLA : aulas

NOMBRE DEL CAMPO	TIPO DE CAMPO	DESCRIPCION
aula_id	int AI PK	Identificador único para cada aula registrado. Clave primaria.
nombre_aula	varchar(100)	Nombre de la aula a registrar.
estatus_aulas_id	int	Identificador único del estatus tipo de dato enteros.

TABLA : calificaciones

NOMBRE DEL CAMPO	TIPO DE CAMPO	DESCRIPCION
calificacion_id	int AI PK	Identificador único para cada calificacion registrada. Clave primaria.
actividad_id	int	Identificador único de la actividad tipo de dato numeros enteros.
no_alumno	varchar(45)	No alumno del cual esta realizando la actividad.
materia_id	int	Identificador único de la materia tipo de dato numeros enteros.
actividad_realizada	text	Comentario de la actividad realizada por parte del alumno.
ruta_archivo	varchar(100)	En este campo solo se inserta la ruta del archivo adjunto.
fecha_envio	timestamp	Este campo se agrega automatica mente una vez que se realiza el inset.
calificacion_actividad	int	Calificacion de la actividad realizada por el alumno.
comentario	text	Comentario de la actividad realizada (retroalimentacion).
id_estatus	int	Identificador único del estatus tipo de dato numeros enteros.

TABLA : concepto

NOMBRE DEL CAMPO	TIPO DE CAMPO	DESCRIPCION
id_concepto	int AI PK	Identificador único para cada concepto registrado. Clave primaria.
Descripcion	varchar(45)	Concepto de pago.

TABLA : estatus

NOMBRE DEL CAMPO	TIPO DE CAMPO	DESCRIPCION
idEstatus	int AI PK	Identificador único para cada estatus registrado. Clave primaria.
Descripcion	varchar(45)	Estatus correspondiente a cada situación.

TABLA : grados

NOMBRE DEL CAMPO	TIPO DE CAMPO	DESCRIPCION
grado_id	int AI PK	Identificador único para cada grado registrado. Clave primaria.
nombre_grado	varchar(100)	Nombre que le corresponde cada grado.
estatus_grados_id	int	Identificador único del estatus tipo de dato numeros enteros.

TABLA : horas

NOMBRE DEL CAMPO	TIPO DE CAMPO	DESCRIPCION
id_horas	int AI PK	Identificador único para cada hora registrado. Clave primaria.
Detalle	time	En este campo se registrarán las horas del día.

TABLA : pagos

NOMBRE DEL CAMPO	TIPO DE CAMPO	DESCRIPCION
id_pagos	int AI PK	Identificador único para cada pago registrado. Clave primaria.
No_alumno	varchar(100)	No alumno registrado.
Nombre_Alumno	varchar(200)	Nombre completo del alumno.
Pago	int	Monto del pago a realizar.
Detalle_mes	varchar(100)	Mes de la mensualidad a pagar.
estatus_id_pago	int	Identificador único del estatus.
concepto_id_pago	int	Identificador único del concepto.
Fecha_registro	timestamp	Este campo se agrega automatica mente una vez que se realiza el inset.

TABLA : materias

NOMBRE DEL CAMPO	TIPO DE CAMPO	DESCRIPCION
materia_id	int AI PK	Identificador único para cada materia registrada. Clave primaria.
nombre_materia	varchar(150)	Nombre de la materia nueva.
dia_semana	varchar(100)	Dia de la semana.
hora_inicio	int	Hora de inicio de la materia
hora_fin	int	Hora fin de la materia.
fecha_creacion	timestamp	Este campo se agrega automatica mente una vez que se realiza el inset.
estatus_materias_id	int	Identificador único del estatus.
grados_grado_id	int	Identificador único del grado.

TABLA : periodos

NOMBRE DEL CAMPO	TIPO DE CAMPO	DESCRIPCION
periodo_id	int AI PK	Identificador único para cada periodo registrado. Clave primaria.
nombre_periodo	varchar(100)	Nombre del periodo a registrar.
estatus_periodos_id	int	Identificador único del estatus.

TABLA : profesor

NOMBRE DEL CAMPO	TIPO DE CAMPO	DESCRIPCION
profesor_id	int AI PK	Identificador único para cada profesor registrado. Clave primaria.
Cedula	varchar(45)	Cedula del profesor.
Nombre	varchar(45)	Nombre del profesor.
Apellido_paterno	varchar(45)	Apellido paterno.
Apellido_Materno	varchar(45)	Apellido materno.
Direccion	varchar(100)	Direccion donde vive actual mente el profesor.
Telefono	varchar(10)	Numero de telefono.
Sexo	varchar(20)	Sexo: Masculino / Femenino.
Fecha_nacimiento	date	Fecha de nacimiento dia/mes/año
Fecha_alta	timestamp	Este campo se agrega automatica mente una vez que se realiza el inset.
estatus_maestro_id	int	Identificador único del estatus.
No_profesor	varchar(45)	No profesor en el formulario se lo asigna automatica mente.

TABLA : users

NOMBRE DEL CAMPO	TIPO DE CAMPO	DESCRIPCION
id	varchar(45)	Identificador único para cada usuario registrado. Clave primaria.
username	varchar(50)	Nombre del usuario con el cual realizara sesión.
password	varchar(255)	Contraseña de la cual estará encriptada.
role	enum('user','admin','profesor')	Tipo de rol para el usuario.
name	varchar(100)	Nombre completo del usuario.
Fecha_creacion	timestamp	Este campo se agrega automática mente una vez que se realiza el insert.

TABLA : profesor\_materia

NOMBRE DEL CAMPO	TIPO DE CAMPO	DESCRIPCION
proceso_id	int AI PK	Identificador único para cada proceso registrado. Clave primaria.
grado_id	int	Identificador único del grado tipo de dato numerico.
aula_id	int	Identificador único de la aula tipo de dato numerico.
profesor_id	int	Identificador único del profesor tipo de dato numerico.
materias_id	int	Identificador único del materia tipo de dato numerico.
periodos_id	int	Identificador único del periodo tipo de dato numerico.
estatus_procesoprofesor_id	int	Identificador único del estatus tipo de dato numerico.



## **TECNOLOGÍAS UTILIZADAS.**

### **HTML**

HTML, siglas de HyperText Markup Language (Lenguaje de Marcado de Hipertexto), es el lenguaje de marcado predominante para la elaboración de páginas web. Es usado para describir la estructura y el contenido en forma de texto, así como para complementar el texto con objetos tales como imágenes. HTML se escribe en forma de "etiquetas", rodeadas por corchetes angulares (<,>).

### **CSS**

Las hojas de estilo en cascada (Cascading Style Sheets, CSS) son un lenguaje formal usado para definir la presentación de un documento estructurado escrito en HTML o XML (y por extensión en XHTML). El W3C es el encargado de formular la especificación de las hojas de estilo que servirá de estándar para los agentes de usuario o navegadores.

### **JAVASCRIPT**

Es un lenguaje interpretado utilizado principalmente en páginas web, con una sintaxis semejante a la del lenguaje Java. Sin embargo, al contrario que Java, JavaScript no es un lenguaje orientado a objetos propiamente dicho, ya que no dispone de herencia. Es más bien un lenguaje basado en prototipos, ya que las nuevas clases se generan clonando las clases base (prototipos) y extendiendo su funcionalidad.

### **PHP Versión 7.4.30**

Es un lenguaje interpretado de propósito general ampliamente usado, diseñado especialmente para desarrollo web y que puede ser incrustado dentro de código HTML. Generalmente se ejecuta en un servidor web, tomando el código en PHP como su entrada y creando páginas web como salida. Puede ser desplegado en la mayoría de los servidores web y en casi todos los sistemas operativos y plataformas sin costo alguno. PHP se encuentra instalado en más de 20 millones de sitios web y en un millón de servidores. Es también el módulo Apache más popular entre las computadoras que utilizan Apache como servidor web.

### **MySQL Workbench 8.0.**

Es un sistema de gestión de base de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones. MySQL AB desarrolla MySQL como software libre en un esquema de licenciamiento dual. Por un lado, se ofrece bajo la GNU GPL para cualquier uso compatible con esta licencia, pero para aquellas empresas que quieran incorporarlo en

productos privativos deben comprar a la empresa una licencia específica que les permita este uso. Está desarrollado en su mayor parte en ANSI C.

## BOOTSTRAP 5

Es un framework CSS y Javascript diseñado para la creación de interfaces limpias y con un diseño responsive. Además, ofrece un amplio abanico de herramientas y funciones, de manera que los usuarios pueden crear prácticamente cualquier tipo de sitio web haciendo uso de los mismos.

## XAMPP

Es un servidor independiente de plataforma, software libre, que consiste principalmente en la base de datos MySQL, el servidor web Apache y los intérpretes para lenguajes de script: PHP y Perl.

## DATATABLES Versión 1.11.0

Es un nuevo tipo de objeto que puede almacenar millones de filas de datos, integrarse con sistemas externos con Data Shuttle y sincronizar subconjuntos de dichos datos en hojas.

DataTable hace que sea posible que use las funciones de Smartsheet para trabajar con grandes cantidades de datos o vistas únicas personalizadas, fórmulas aplicadas, flujos de trabajo y más en dichos datos. Y sus datos siempre son uniformes y están actualizados.

## TEMPLATE VOLER

Voler es una plantilla HTML5 de panel de administración gratuita creado usando Bootstrap. Diferentes características se han utilizado en la plantilla incluyendo estadísticas, formulario de contacto, Google Maps, Páginas especiales y efectos de desplazamiento. Utiliza la biblioteca JQuery JavaScript y se basa en la biblioteca bootstrap5. La plantilla es 100% lista para dispositivos móviles, compatible con navegadores y retina-friendly.

## JQUERY

jQuery es una biblioteca o framework de JavaScript que permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la tecnología AJAX a páginas web.

## CODIGO DEL PROYECTO

Todo el proyecto será entregado en un dispositivo de almacenamiento al personal correspondiente.

