



**UNIVERSIDAD NACIONAL AUTÓNOMA DE
MÉXICO**

**FACULTAD DE CONTADURÍA Y
ADMINISTRACIÓN**

SITIO WEB PARA EL CONTROL ESCOLAR DE UNA
INSTITUCIÓN PÚBLICA DE EDUCACIÓN BÁSICA DE NIVEL
PRIMARIA

DISEÑO DE UN SISTEMA O PROYECTO

CESAR BADILLO BARCENAS



México Cd. Mx.

2021



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE CONTADURÍA Y ADMINISTRACIÓN

SITIO WEB PARA EL CONTROL ESCOLAR DE UNA
INSTITUCIÓN PÚBLICA DE EDUCACIÓN BÁSICA DE NIVEL
PRIMARIA

DISEÑO DE UN SISTEMA O PROYECTO
QUE PARA OBTENER EL TÍTULO DE:
LICENCIADO EN INFORMÁTICA

PRESENTA:

CESAR BADILLO BARCENAS

ASESOR:

DRA. GRACIELA BRIBIESCA CORREA



Índice general

Introducción	1
Problemática	2
Objetivo general	3
Objetivos específicos.....	3
Aportaciones, consideraciones y limitaciones en este trabajo de titulación.....	4
Resumen capitular.....	4
Capítulo 1. Marco teórico.....	4
Capítulo 2. Generalidades de la institución.....	4
Capítulo 3. Proceso del software.	5
Capítulo 4. Análisis y requerimientos.....	5
Capítulo 5. Diseño.	5
Capítulo 6. Desarrollo.....	5
CAPÍTULO 1. MARCO TEÓRICO.....	6
1.1 La educación básica en el mundo.....	7
1.2 El derecho a la educación en México	8
1.3 Sistema	8
1.4 Sistema de información.....	9
1.5 Sistema tecnológico de información	9
1.6 Tecnología de la información (TI).....	10
1.7 Aplicaciones web.....	10
1.8 Bases de datos.....	10
1.9 El modelo de datos relacional	12
CAPÍTULO 2. GENERALIDADES DE LA INSTITUCIÓN	13
2.1 Características de la institución.....	14
2.2 Visión	15
2.3 Misión.....	15
2.4 Organigrama.....	16
2.5 Objetivos de la institución	16
CAPÍTULO 3. PROCESO DEL SOFTWARE	17
3.1 Metodologías de software	18
3.2 Análisis de software	19
3.3 Diseño de software.....	20

3.4 Implementación del software	20
3.5 Pruebas del software.....	21
3.6 Mantenimiento.....	21
3.7 Documentación	22
CAPÍTULO 4. ANÁLISIS	23
4.1 Análisis de requerimientos.....	24
4.2 Requerimientos de usuario	25
4.3 Requerimientos del sistema.....	26
4.4 Diagramas de casos de uso	26
4.5 Diagramas de secuencia	30
4.6 Secuencia de acciones para los casos de uso identificados	31
4.7 Descripción de escenarios de casos de uso.....	37
CAPÍTULO 5. DISEÑO	48
5.1 Arquitectura cliente – servidor.....	49
5.2 Diseño de la interfaz de usuario.....	50
5.3 Diseño conceptual de la base de datos	54
5.4 Diccionario de datos.....	59
5.5 Diagramas de flujo.....	65
5.6 Diagramas de flujo de los procesos del sistema de control escolar.....	67
CAPÍTULO 6. DESARROLLO	79
6.1 Servidor web.....	80
6.2 PHP	80
6.3 HTML	80
6.4 MYSQL	82
6.5 MariaDB.....	82
6.6 Servidor de pruebas locales con XAMPP	82
6.7 Validación de los campos de formularios.....	83
6.9 Creación de las tablas que conforman la base de datos.	84
6.10 Conexión a MySQL desde PHP	88
6.11 Sesiones para acceso a los usuarios.	90
6.12 Protocolo IP	93
6.13 Dirección IP.....	93
6.14 Servicio de identificación de dominios.....	94
6.15 Servicio de web hosting.....	94

6.16 Administración del sitio.....	95
Conclusiones.....	97
Anexos.....	99
Manuales de usuario.....	99
Manual de usuario para el tutor o padre de familia.....	99
Manual de usuario para el docente.....	104
Manual de usuario para el administrador.....	112
Bibliografía.....	118

Índice de cuadros

Cuadro 4.1 Caso de uso tutor.....	27
Cuadro 4.2 Casos de uso docente.....	28
Cuadro 4.3 Casos de uso administrador.....	29
Cuadro 4.4 Descripción del caso de uso “ingresar datos del alumno y tutor”.....	37
Cuadro 4.5 Descripción del caso de uso “registrar calificaciones trimestrales”.....	38
Cuadro 4.6 Descripción del caso de uso “actualizar o modificar calificaciones”.....	39
Cuadro 4.7 Descripción del caso de uso “consultar calificaciones”.....	40
Cuadro 4.8 Descripción del caso de uso “consultar datos del alumno”.....	41
Cuadro 4.9 Descripción del caso de uso “consultar datos del tutor”.....	42
Cuadro 4.10 Descripción del caso de uso “consultar datos de alumnos y tutores”.....	43
Cuadro 4.11 Descripción del caso de uso “actualizar datos de alumnos y tutores”.....	44
Cuadro 4.12 Descripción del caso de uso “consultar calificaciones por grupo”.....	45
Cuadro 4.13 Descripción del caso de uso “agregar nuevos alumnos”.....	46
Cuadro 4.14 Descripción del caso de uso “eliminar alumnos”.....	47
Cuadro 5.1 Tabla alumnos_base.....	59
Cuadro 5.2 Tabla alumnos_inscritos.....	62
Cuadro 5.3 Tabla tutores.....	65
Cuadro 6.1 Etiquetas HTML básicas.....	81

Índice de figuras

Figura 2.1 Fachada del centro escolar.....	14
Figura 2.2 Auditorio del Centro Escolar Nezahualcóyotl.....	15
Figura 2.3 Organigrama de la institución	16
Figura 4.1 Diagrama de caso de uso.....	26
Figura 4.2 Diagrama de casos de uso tutor	27
Figura 4.3 Diagrama de casos de uso docente	28
Figura 4.4 Diagrama de casos de uso administrador	29
Figura 4.5 Diagrama UML de secuencia de acciones	30
Figura 4.6 Secuencia de acciones para los casos de uso “Ingresar datos del alumno” e “Imprimir comprobante”	31
Figura 4.7 Secuencia de acciones para el caso de uso “registrar calificaciones”	31
Figura 4.8 Secuencia de acciones para el caso de uso “actualizar calificaciones”	32
Figura 4.9 Secuencia de acciones para el caso de uso “consultar calificaciones”	32
Figura 4.10 Secuencia de acciones para los casos de uso “consultar datos de alumnos”	33
Figura 4.11 Secuencia de acciones para los casos de uso “consultar datos de tutores”	33
Figura 4.12 Secuencia de acciones para los casos de uso “consultar datos de alumnos y tutores”	34
Figura 4.13 Secuencia de acciones para el caso de uso “actualizar datos de alumnos y tutores”	34
Figura 4.14 Secuencia de acciones para el caso de uso “consultar calificaciones trimestrales por grupo”	35
Figura 4.15 Secuencia de acciones para el caso de uso “agregar nuevos alumnos”	35
Figura 4.16 Secuencia de acciones para el caso de uso “eliminar alumnos”	36
Figura 5.1 El modelo de 3 capas.....	50
Figura 5.2 Interfaces de usuario para el tutor.....	51
Figura 5.3 Interfaces de usuario para el docente.....	52
Figura 5.4 Interfaces de usuario para el administrador	53
Figura 5.5 Representación de una entidad fuerte	54
Figura 5.6 Representación de una entidad débil	55
Figura 5.7 Representación de una relación de entidades	55
Figura 5.8 Representación de atributos de una entidad.....	56
Figura 5.9 Modelo conceptual para el sistema de control escolar	57
Figura 5.10 Diagrama relacional para el sistema de control escolar	58
Figura 5.11 Inicio o fin	65
Figura 5.12 Proceso	65
Figura 5.13 Decisión	66

Figura 5.14 Subrutina	66
Figura 5.15 Conectores.....	66
Figura 5.16 Diagrama de flujo del proceso de captura de datos del alumno y tutor	67
Figura 5.17 Diagrama de flujo del proceso de captura de calificaciones.....	68
Figura 5.18 Diagrama de flujo del proceso de consulta de calificaciones.....	69
Figura 5.19 Diagrama de flujo del proceso de actualización de calificaciones	70
Figura 5.20 Diagrama de flujo del proceso de consulta de datos del alumno	71
Figura 5.21 Diagrama de flujo del proceso de consulta de datos del tutor	72
Figura 5.22 Diagrama de flujo del proceso de consulta de datos del alumno	73
Figura 5.23 Diagrama de flujo del proceso de consulta de datos del tutor	74
Figura 5.24 Diagrama de flujo del proceso de actualización de datos de alumnos o tutores	75
Figura 5.25 Diagrama de flujo del proceso de consulta calificaciones por grupo.....	76
Figura 5.26 Diagrama de flujo del proceso de alta de nuevo alumno	77
Figura 5.27 Diagrama de flujo del proceso de eliminación de un alumno.....	78
Figura 6.1 Pruebas en el servidor local XAMPP.....	83
Figura 6.2 Acceso para el administrador	90
Figura 6.3 Cliente FTP FileZilla	96
Figura A.1 Página de inicio para el tutor	99
Figura A.2 Página de búsqueda de clave CURP	100
Figura A.3 Página con resultado de búsqueda	101
Figura A.4 Iniciar proceso de inscripción.....	101
Figura A.5 Ejemplo de llenado de formulario para el tutor	102
Figura A.6 Envío del formulario.....	103
Figura A.7 Botón imprimir comprobante	103
Figura A.8 Ejemplo un comprobante de inscripción	104
Figura A.9 Página de inicio para el docente	105
Figura A.10 Página de consulta de datos de alumnos y tutores	106
Figura A.11 Botones para acceso a datos de alumno y tutor.....	106
Figura A.12 Página con información de un alumno	107
Figura A.13 Salir del sistema	108
Figura A.15 Captura de calificaciones no habilitada	109
Figura A.16 Botón para actualizar calificaciones.....	109
Figura A.17 Página de captura de calificaciones	110
Figura A.18 Botón guardar cambios de calificaciones.....	111

Figura A.19	Página de inicio del administrador.....	112
Figura A.20	Consulta de datos de alumnos y tutores.....	113
Figura A.21	Ejemplo de consulta de datos de un alumno.....	113
Figura A.22	Botones de actualización de datos de alumnos y tutores	114
Figura A.23	Formulario de edición de datos	114
Figura A.24	Consulta de calificaciones por grupo	115
Figura A.25	Ingresar un nuevo alumno	116
Figura A.26	Botones de edición y eliminación de registros	117

Dedicatorias

Dedico el presente trabajo a los amores de mi vida, mis tres hermosas niñas:

Paola, Denisse y Yamilet.

También a mi esposa Teresita, por su apoyo incondicional en todos mis proyectos, y por estar siempre a mi lado, en todo lo que decido emprender.

Agradecimientos

El siguiente trabajo de titulación no hubiera sido posible sin la colaboración de múltiples instancias y personas, que contribuyeron con su apoyo, conocimientos, opiniones y valiosas sugerencias para cumplir con los objetivos que se plantearon al inicio de este proyecto.

Agradecimientos infinitos:

A la Universidad Nacional Autónoma de México, que a través de la Facultad de Contaduría y Administración me ha brindado los conocimientos que me guiarán durante toda mi carrera profesional.

A la Dra. Graciela Bribiesca Correa, por sus valiosos y acertados comentarios para la redacción y estructura durante el desarrollo del presente trabajo de titulación.

Al Centro Escolar Nezahualcóyotl, incluyendo directivos, personal docente y sociedad de padres de familia por su apoyo, disposición y tiempo otorgado durante el desarrollo de este proyecto.



Introducción

Desde siempre, los grupos humanos han utilizado simultáneamente diversos tipos de prácticas y actividades sociales con el fin de facilitar a las nuevas generaciones el acceso a las formas y saberes culturales, conocimientos y creencias sobre el mundo; lenguaje y otros instrumentos simbólicos utilizados para comunicarse, para conocer la realidad, representarla y actuar sobre ella; tecnologías y técnicas; tradiciones; sistemas de valores; etcétera, considerados fundamentales para la supervivencia colectiva y cuya apropiación individual se juzga necesaria, en consecuencia, para llegar a formar parte de ellos como miembros de pleno derecho. Desde siempre y en todo tipo de sociedades, la responsabilidad en la organización y conducción de estas prácticas y actividades sociales ha estado a cargo de diferentes actores, y a menudo también de diferentes instancias, que, desde su rol específico y su ubicación en la organización social, cultural y económica establecida, contribuyen conjuntamente al desarrollo personal y a la socialización de las nuevas generaciones.

El derecho a la educación prioriza el interés superior de la niñez y de la adolescencia en el acceso, la permanencia y la participación en los servicios educativos. Ante la realidad de pobreza y desigualdad de nuestro país, la reciente reforma constitucional instituye la equidad como el principio redistributivo a fin de asegurar progresivamente el ejercicio pleno de este derecho para todos los niños y jóvenes.

El Centro Escolar Nezahualcóyotl, es una institución pública de nivel primaria, que tiene su fundación el 06 de julio de 1951, se encuentra ubicado en el centro de la comunidad de Texcoco, Estado de México, goza de gran prestigio entre en la localidad en que encuentra situada y las comunidades aledañas, situación que la coloca como la escuela más importante en su nivel, dentro del municipio en que se inserta. Atiende una matrícula aproximada de 1400 estudiantes durante el ciclo escolar, cada uno de los seis grados está dividido a su vez en seis grupos, conformando un total de 36 grupos con un promedio de más de 40 alumnos en cada uno de ellos.

Además de ofrecer las asignaturas básicas del plan de estudios nacional para la educación primaria, el Centro Escolar Nezahualcóyotl, brinda además asignaturas extracurriculares complementarias como son la enseñanza del idioma inglés y habilidades informáticas.



Problemática

En particular, esta institución debe matricular cada ciclo escolar a todos los alumnos que ingresan a un nuevo grado de educación primaria, lo que representa una tarea complicada debido a la gran cantidad de estudiantes que hay en el plantel y a que todo el proceso se lleva a cabo de forma manual, es decir, sin utilizar ninguna herramienta tecnológica que ayude a que dicho proceso sea más rápido y preciso.

En consecuencia, se derivan una serie de dificultades que a continuación se describen:

- Lentitud en el proceso. Los padres de familia hacen largas filas de espera para ser atendidos por el personal docente de la escuela.
- Información faltante e imprecisa. En numerosas ocasiones los padres o tutores de familia no completan la información requerida en los formularios.
- Consultas ineficientes. Cuando la dirección escolar necesita consultar información acerca de algún alumno o padre de familia, se invierte un tiempo considerable, debido a que dicha información se encuentra almacenada en expedientes de papel, y la búsqueda debe hacerse de forma manual.
- Uso excesivo de papel. Se requiere de una cantidad importante de papel en el proceso, ya que se utilizan formularios, avisos, listas, etcétera.

Otro problema que no es derivado de la situación anterior, pero que es importante mencionar, es que debido a la pandemia de COVID-19 en el mundo, el ciclo escolar 2020 – 2021 de educación básica en México al igual que en otros países, ha tenido que iniciar completamente a distancia, dificultando la recolección de los datos de alumnos y sus tutores de la forma convencional, es decir, reuniendo presencialmente a los padres de familia en la institución para realizar su trámite de inscripción.

Observando dicha problemática, se propone implementar una solución informática para la automatización del proceso de inscripción, concretamente se plantea desarrollar un sitio web que permita en primera instancia llevar a cabo el registro de los datos de alumnos y tutores, y que facilite además la consulta de información por parte de la dirección escolar en



el momento que sea necesario. Se propone además que el sitio web facilite a los docentes de cada grupo realizar el reporte de sus calificaciones trimestrales para que estén siempre disponibles y puedan ser consultadas a su vez por la dirección escolar.

Por otro lado, es importante señalar un beneficio adicional que conlleva utilizar una aplicación web para llevar a cabo los procesos anteriormente mencionados, porque al no utilizar papel en el proceso, se favorece la preservación del medio ambiente, ya que se reduce drásticamente la papelería utilizada.

En consecuencia y considerando la problemática anterior, se propone el siguiente objetivo.

Objetivo general

Diseñar, desarrollar e implementar una aplicación web con motor de base de datos que permita el control de la información alineada a los objetivos del Centro Escolar Nezahualcóyotl ubicado en el municipio de Texcoco, para el ciclo escolar 2020 – 2021.

Objetivos específicos

- Identificar los requerimientos del sistema y de los usuarios involucrados.
- Diseñar una base de datos para el almacenamiento de la información perteneciente a los alumnos y tutores, así como también sus calificaciones trimestrales.
- Diseñar interfaz web intuitiva y amigable para todos los usuarios del sitio web.
- Diseñar los algoritmos para llevar a cabo cada uno de los procesos requeridos.
- Escribir el código fuente de cada una de las páginas que conformarán el sitio web, así como la creación física de la base de datos y sus tablas.
- Realizar pruebas en la aplicación introduciendo datos simulados y reales.



- Configurar y desplegar el sitio web en un servicio de hosting profesional para garantizar el acceso y la disponibilidad las 24 horas del día.

Aportaciones, consideraciones y limitaciones en este trabajo de titulación.

El sitio web para el control escolar provee una solución para automatizar el proceso de inscripción, poniendo a disposición una dirección electrónica en donde se lleva a cabo el registro de cada uno de los datos pertenecientes a los alumnos y tutores, para ser almacenados en una base de datos permanente que puede ser consultada posteriormente por la dirección escolar. Además, facilita el registro de calificaciones trimestrales por parte de los docentes encargados de cada grupo.

Cabe aclarar que esta solución web, aunque resuelve la problemática principal anteriormente mencionada, no puede ser considerada como una solución integral para el control escolar de la institución, ya que existen otras áreas de oportunidad que podrían resolverse implementando mejoras y módulos adicionales quizás en un futuro cercano.

Resumen capitular

Capítulo 1. Marco teórico.

Ofrece un panorama general de los conceptos más importantes abordados en este proyecto, tales como la educación básica en México y el mundo, así como también nociones básicas de los sistemas de información.

Capítulo 2. Generalidades de la institución

Se describen las características más importantes de la institución, así como aspectos importantes de su organización: Misión, visión, objetivos y organigrama.



Capítulo 3. Proceso del software.

Refiere las fases típicas del desarrollo de software haciendo énfasis en el modelo lineal o en cascada.

Capítulo 4. Análisis y requerimientos.

Se aborda a detalle la problemática de la organización, se particularizan las necesidades de los usuarios y se describen conceptos importantes para la correcta especificación de requerimientos de usuario y del sistema, utilizando para ello, diagramas de casos de uso y de secuencia.

Capítulo 5. Diseño.

Se describe el diseño de la aplicación basándose en las especificaciones de la etapa de requerimientos, se define la arquitectura de la aplicación y se diseñan las páginas para la interfaz gráfica de los usuarios. Además, se lleva a cabo el diseño conceptual y lógico de los datos que formarán parte del sistema.

Capítulo 6. Desarrollo.

Se abordan los aspectos más importantes en esta fase del desarrollo, se presentan algunas muestras del código fuente utilizado, así como las herramientas informáticas utilizadas y conceptos básicos importantes para comprender mejor cómo se llevó a cabo la implementación de la aplicación web.



CAPÍTULO 1. MARCO TEÓRICO



1.1 La educación básica en el mundo

La educación ha sido considerada siempre en los grupos humanos una responsabilidad compartida por diferentes actores que, en el marco de las actividades habituales y de los escenarios en los que estas actividades se desarrollan, asumen conjuntamente el compromiso de satisfacer las necesidades educativas de todos sus miembros, y en especial de los miembros más jóvenes y de los recién llegados, ofreciéndoles la oportunidad de participar en dichas actividades y en dichos escenarios.

La educación nos dota de los conocimientos, valores y aptitudes que constituyen la base del éxito educativo y profesional a lo largo de toda la vida. Una educación de calidad debe tener como objetivo primordial la infancia, tener en cuenta las cuestiones de género y adaptarse a las necesidades de los distintos grupos de edad. Debe basarse en un programa de estudios que tenga en cuenta las necesidades y la realidad de todos los estudiantes, y emplear como vehículo de transmisión a profesores formados en la docencia y asistidos del material didáctico adecuado. Los entornos adecuados para la infancia son seguros, limpios, y favorecen el que los niños y niñas jueguen y aprendan.

Tradicionalmente, los esfuerzos para impulsar el acceso a la educación se han centrado en los años de la escuela primaria, un periodo fundamental para la adquisición de conocimientos alfabéticos, de aritmética elemental y otras destrezas básicas.

Los niños y niñas comienzan su aprendizaje en el momento de nacer, y sus primeros años de vida son críticos para el desarrollo de sus habilidades cognitivas, lingüísticas, emocionales y sociales. En consecuencia, los programas de desarrollo de la primera infancia constituyen un paso fundamental para la mejora de las sociedades y para la consecución de la igualdad entre los géneros.



1.2 El derecho a la educación en México

El artículo 3º de la Constitución Política de los Estados Unidos Mexicanos establece la educación como un derecho humano. Su ejercicio es el principal mecanismo social del que dispone el Estado mexicano para desarrollar la formación integral de las facultades de sus ciudadanos. Esta función social de la educación, entre otros aspectos, coadyuva al desarrollo de la democracia, a la comprensión y la solución de nuestros problemas con una perspectiva nacional, así como a una mejor convivencia humana. La educación será integral y desarrollará capacidades cognitivas, socioemocionales y físicas que preparen para la vida todas las personas; será inclusiva, intercultural y garantizará los aprendizajes de todos los alumnos para el desarrollo de un pensamiento crítico.

1.3 Sistema

Un sistema puede definirse como un sistema coherente de componentes interdependientes que existe con algún objetivo, tiene cierta estabilidad y puede ser visto fácilmente como un todo. Generalmente los sistemas se describen mediante un modelo existente de entrada, proceso y salida, dentro de un determinado entorno. El entorno de un sistema puede definirse como cualquier cosa ajena al sistema, pero que afecta la forma en la que opera el propio sistema. Las entradas al sistema son los recursos que obtiene de su entorno o de otros sistemas. Las salidas del sistema son aquellas cosas que éste proporciona de vuelta a su entorno o a otros sistemas. El proceso es la actividad mediante la cual el sistema transforma sus entradas en sus salidas. El concepto de sistema se aplica tanto a la tecnología como a la actividad humana.



1.4 Sistema de información

Un sistema de información es un sistema de comunicación entre las personas. Los sistemas de información son sistemas implicados en la recogida, tratamiento, distribución y uso de la información. Los sistemas de información prestan apoyo a los sistemas de actividad humana.

1.5 Sistema tecnológico de información

Un sistema tecnológico de información es una colección organizada de hardware, software y tecnología de las comunicaciones y los datos diseñada para sustentar ciertos aspectos de un determinado sistema de información. Las entradas de un sistema tecnológico de información son datos, realiza alguna manipulación sobre ellos como parte del proceso y devuelve los datos manipulados para que sean interpretados dentro de algún sistema de actividad humana.

Desde el punto de vista informático, toda organización es un sistema de información formados por un conjunto de recursos técnicos, humanos y económicos, organizados e interrelacionados de forma activa. El objetivo del sistema de información no es otro que el de dar respuesta a un tratamiento automatizado de la información que maneja la organización posibilitando el funcionamiento y la comunicación en la misma, para tal fin se emplean como soporte las tecnologías de la información y las comunicaciones. En el complejo mundo moderno de las organizaciones, la mayoría de los sistemas de información dependen del hardware, el software y de la tecnología de las comunicaciones y los datos en mayor o menor grado, ya que la eficiencia y la eficacia aumentan cuando se utiliza esta tecnología. Resumiendo, un sistema tecnológico de información está compuesto por los siguientes elementos:

Datos. Es la información relevante que almacena y gestiona el sistema de información, los datos por sí solos no aportan conocimiento, ya que es necesario procesarlos y transformarlos para convertirlos en información.



Hardware. Es el equipo físico que se utiliza para el manejo de los datos, facilita la comunicación, el procesamiento y el almacenamiento de la información.

Software. Son las aplicaciones que permiten el funcionamiento adecuado del sistema.

Recursos humanos. Es el personal que opera el sistema de información, al introducir, manipular y consultar los datos en función de sus actividades establecidas.

1.6 Tecnología de la información (TI)

La tecnología de la información es una tecnología que se utiliza para sustentar la recogida, tratamiento, distribución y uso de la información. La tecnología de la información proporciona medios para construir algunos aspectos de los sistemas de información, pero es diferente de los sistemas de información. La moderna tecnología de la información se compone de hardware, software y tecnología de las comunicaciones y datos.

1.7 Aplicaciones web

Las aplicaciones web utilizan lo que se conoce como clientes livianos los cuales no ejecutan excesivas labores de procesamiento para ejecutar la aplicación misma. Desde el punto de vista de la arquitectura del software se diferencian dos partes; por un lado, el cliente, donde el usuario final utiliza la aplicación a través de un navegador o browser de internet, por medio de este cliente web, el usuario interactúa con la aplicación que se ubica del otro lado, es decir, en el servidor, que es donde se encuentran realmente los datos y la lógica de negocios de la aplicación.

1.8 Bases de datos

Una base de datos puede definirse como una colección o depósito de datos integrados con redundancia controlada y con una estructura que refleje las interrelaciones y restricciones existentes en el mundo real. Los datos que serán compartidos por diferentes usuarios y aplicaciones deben mantenerse independientes de estas, y su definición y descripción únicas para cada tipo de dato, deben estar almacenadas junto con los mismos. Los



procedimientos de actualización y recuperación deben ser capaces de conservar la seguridad, integridad confidencialidad, y disponibilidad del conjunto de datos.

Un sistema gestor de base de datos o DBMS por sus siglas en inglés *Data Base Management System*, es el software que permite a los usuarios procesar, describir, administrar y recuperar los datos almacenados en una base de datos. En estos sistemas se proporciona un conjunto de herramientas que permiten a los diferentes usuarios realizar sus labores habituales con los datos, garantizando la seguridad de los mismos.

El objetivo principal de un sistema de base de datos es la de proporcionar a los usuarios que tienen acceso al sistema, una visión de los datos de una manera abstracta, sin visualizar informaciones acerca del almacenamiento, además deben:

- Facilitar el almacenamiento de grandes cantidades de columnas de datos.
- Recuperar de forma rápida y sencilla la información o los datos.
- Facilitar la organización y estructuración de los datos.
- Mejorar la distribución de la información a través de diversos medios.

A su vez, un sistema de base de datos debe eliminar o disminuir los siguientes problemas:

Redundancia e inconsistencia de datos. Está presente cuando se van repitiendo datos que son parte de los archivos que forman una base de datos, en la cual se irán almacenando de forma repetida en diferentes sitios. Un sistema de base de datos estará más expuesto a fallos si existen datos repetidos.

Inconsistencia de datos. Se origina cuando los datos redundantes o repetidos no sean iguales entre sí, es decir cuando se modifica o actualiza un dato en un lugar, sin que se produzca ningún cambio o actualización en su duplicado.

Aislamiento de los datos. En una base de datos los datos son independientes de los tratamientos que se realicen sobre ellos, esto significa que, aunque se modifiquen los programas que actúan sobre los datos, estos no sufrirán ninguna modificación.

Anomalías en el acceso concurrente. Cuando una base de datos es consultada por múltiples usuarios simultáneamente, se debe evitar que existan anomalías que pueden surgir cuando un usuario se encuentra modificando el registro de una tabla y otro usuario intente realizar alguna operación en ese mismo registro.



Seguridad. Deben existir mecanismos de protección de visibilidad, creación, modificación o eliminación de los datos dependiendo de los permisos para cada usuario que accede al sistema.

1.9 El modelo de datos relacional

El modelo de datos relacional constituye un marco formal de representación de la estructura de los datos basado en el álgebra de conjuntos y la lógica.

Características básicas del modelo de datos relacional

Los elementos estructurales son las relaciones, los atributos y las tuplas, respectivamente, popularmente conocidas gracias al lenguaje SQL como tablas, columnas y filas. Las relaciones agrupan una serie de atributos que denotan el esquema de la relación. El conjunto de tuplas de cada relación conforma la extensión de la relación.

Sobre estos elementos se define un conjunto de operaciones procedimentales, que da lugar al álgebra relacional. Las principales operaciones son la unión, intersección, diferencia, el producto cartesiano, la selección, la proyección y la combinación. Las operaciones de selección, proyección y combinación son operaciones específicas del modelo de datos relacional, mientras que la unión, intersección, diferencia, y el producto cartesiano pertenecen al álgebra de conjuntos.

También se proporciona un lenguaje declarativo de alto nivel, basado en el cálculo de predicados, el lenguaje SQL, que contiene un conjunto de sentencias que permiten crear y manipular Bases de datos relacionales.

Entendemos el modelo relacional como una propuesta de ver los datos como si se trataran de objetos del mundo real, diferenciables entre sí por sus características básicas. Un objeto dado puede ser descrito por la colección de características que tiene, denominadas atributos. Este modelo admite relaciones uno a varios, uno a uno, y varios a varios.



CAPÍTULO 2. GENERALIDADES DE LA INSTITUCIÓN

2.1 Características de la institución

El Centro Escolar Nezahualcóyotl, es una de las instituciones públicas de educación primaria más importantes de la región, dentro de las características que la rodean, podemos citar que alberga una población de las comunidades circunvecinas de la cabecera municipal, además de recibir a niños provenientes de municipios vecinos como: San Vicente Chicoloapan, Chiconcuac, Papalotla, Tepetlaoxtoc, Atenco, Chiautla y Tezoyuca. Cabe destacar que estos municipios a pesar de tener sus diferencias en la manera de conducirse socialmente, dada su cercanía e historia, poseen características similares a las de las comunidades del municipio de Texcoco.



Figura 2.1 Fachada del centro escolar

Con respecto a la infraestructura para la enseñanza de habilidades informáticas, la escuela cuenta con dos talleres de cómputo conformados por 25 equipos de escritorio cada uno, y conectados permanentemente a internet. Además, también cuenta con un auditorio con una capacidad aproximada de 300 personas, donde se llevan a cabo diferentes eventos como obras de teatro, presentaciones artísticas, talleres, conferencias, etcétera.



Figura 2.2 Auditorio del Centro Escolar Nezahualcóyotl

2.2 Visión

Somos una institución educativa de calidad comprometida a desarrollar de manera profesional en sus alumnos los conocimientos, habilidades, actitudes y valores que les permitirán integrarse a la sociedad actual.

2.3 Misión

Ofrecer un servicio educativo de calidad que nos permita lograr el perfil de egreso de los alumnos; poniendo énfasis en la formación de valores, la toma de decisiones, la autorregulación y la cultura de la no violencia, así como en las habilidades comunicativas y en el trabajo en equipo. ello mediante el desarrollo de competencias intelectuales para la comprensión de la lectura, la selección de la información, la expresión oral y escrita, la adquisición de razonamiento matemático para aplicarlo en la solución de problemas cotidianos, así como también inculcar conocimientos científicos básicos y valores fundamentales para comprender el medio social y natural, preservar la salud y el medio ambiente, mejorar la convivencia social y disfrutar las artes y el ejercicio físico.

A continuación, se puede apreciar la estructura organizativa de la escuela a través del siguiente organigrama.



2.4 Organigrama

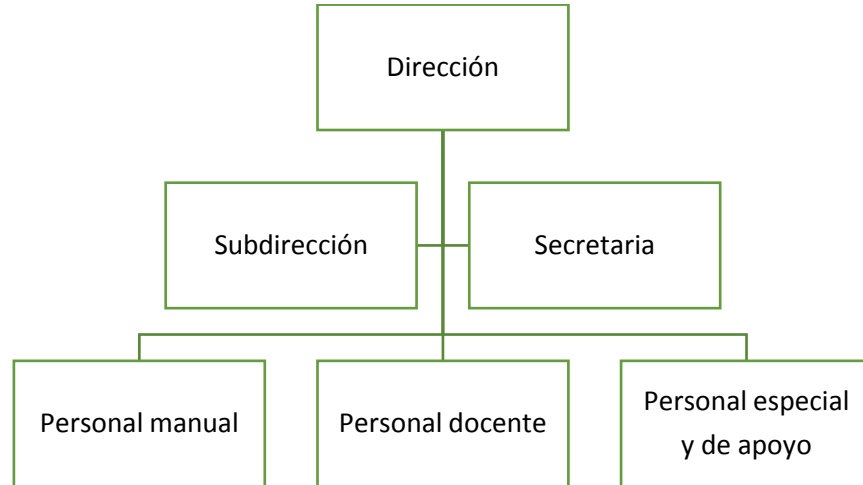


Figura 2.3 Organigrama de la institución

2.5 Objetivos de la institución

Proporcionar a los alumnos y las alumnas los aprendizajes de la expresión y comprensión oral, la lectura, la escritura, el cálculo, la adquisición de nociones básicas de la cultura, y el hábito de convivencia así como los de estudio y trabajo, el sentido artístico, la creatividad, afectividad y acondicionamiento físico, con el propósito de garantizar una formación integral que contribuya al pleno desarrollo de la personalidad de los alumnos y las alumnas y de prepararlos para cursar con aprovechamiento la Educación Secundaria.



CAPÍTULO 3. PROCESO DEL SOFTWARE



3.1 Metodologías de software

Una metodología es un marco de trabajo que puede ser utilizado como guía de las actividades que se van a llevar a cabo, permite organizar el proceso de desarrollo de software a través de la definición de pautas a seguir y restricciones a cumplir. Existen diferentes modelos de procesos como son:

- El modelo en cascada. Es un modelo de desarrollo lineal o secuencial. El proyecto de software es dividido en fases que deben procederse en forma secuencial.
- Desarrollo incremental. La base de este modelo es el modelo lineal o secuencial que se repite iterativamente, en cada iteración se incrementa la funcionalidad al producto hasta que se implementan todos los requerimientos.
- Desarrollo en espiral. Este modelo es una combinación de los modelos en cascada e iterativo. El proyecto se divide en proyectos más pequeños los cuales evoluciona el producto.

El modelo en cascada, describe las actividades fundamentales del desarrollo de software de la siguiente manera:

1. Análisis y definición de requerimientos. Los servicios, restricciones y metas del sistema se definen a partir de las consultas con los usuarios. Entonces se definen en detalle y sirven como una especificación del sistema.
2. Diseño del sistema y del software. El proceso del diseño del sistema divide los requerimientos en sistemas hardware y software. Establece una arquitectura completa del sistema. El diseño del software identifica y describe las abstracciones fundamentales del sistema software y sus relaciones.
3. Implementación y prueba de unidades. Durante esta etapa, el diseño del software se lleva a cabo como un conjunto o unidades de programas. La prueba de unidades implica verificar que cada uno cumpla su especificación.
4. Integración y prueba del sistema. Los programas o las unidades individuales de programas se integran y prueban como un sistema completo para asegurar que se cumplan los requerimientos del software. Después de las pruebas, el sistema software se entrega al cliente.



5. Funcionamiento y mantenimiento. El sistema se instala y se pone en funcionamiento práctico. El mantenimiento implica corregir errores no descubiertos en las etapas anteriores del ciclo de vida, mejorar la implementación de las unidades del sistema y resaltar los servicios del sistema una vez que se descubren nuevos requerimientos.

Ventajas del modelo en cascada

La ventaja de utilizar el modelo en cascada es que los riesgos son descubiertos tempranamente debido a que los requerimientos no cambian al inicio. Esto permite tener una apreciación muy aproximada acerca del tiempo y los costos necesarios.

Desventajas del modelo en cascada

Por la misma razón de que los requerimientos no cambian, se dificulta llevar a cabo cualquier modificación. Además, debido a que una fase depende por completo de la otra, el fallo en alguna de ellas podría reflejarse en todas las demás.

3.2 Análisis de software

La fase de análisis de software define los requisitos del software que se pretende desarrollar, el analista de software es el profesional responsable de identificar el problema o necesidad que se quiere resolver y obtener los requisitos para crear una solución.

El conjunto de requisitos define lo que el sistema debe hacer para satisfacer las necesidades identificadas. En esta fase se debe establecer una buena comunicación con el cliente y con los usuarios finales para tener una idea general de lo que se necesita.

Una vez obtenidos los requisitos, el analista continúa con el proceso de análisis identificando de una manera abstracta las funcionalidades que el sistema deberá tener para cumplir con los requisitos.

Una práctica muy frecuente para mapear cada funcionalidad es a través de los casos de uso, que es una especie de descripción de lo que sucede entre el usuario y el sistema. También es muy común elaborar diagramas UML (Lenguaje Unificado de Modelado) por sus siglas en inglés, para representar los casos de uso de una manera visual.



3.3 Diseño de software

En esta fase se pretende determinar el funcionamiento de una forma general y sin muchos detalles. Los objetivos principales son establecer las consideraciones de los recursos del sistema físicos y lógicos. Tomando como base la información obtenida en la fase de análisis, los arquitectos y desarrolladores proponen una solución tecnológica al problema. Este diseño normalmente se lleva a cabo sin definir todavía qué tecnologías, frameworks y bibliotecas serán utilizadas, se pueden utilizar los siguientes diagramas UML en esta fase.

- Diagrama de componentes. Representa la división en alto nivel de los principales componentes del sistema, mostrando la división lógica de las responsabilidades de cada componente.
- Diagrama de despliegue. Es una representación del entorno en donde se ejecutará el sistema, como servidores, bases de datos, proxies, replicación, etcétera.
- Diagrama de secuencia. Representa una acción determinada en el sistema, mostrando la interacción entre diversos objetos a través de las llamadas ejecutadas y de retorno, lo que permite observar la secuencia de llamadas a través del tiempo.

3.4 Implementación del software

En esta fase se debe seguir lo que se ha definido en el diseño, los programadores tienen la libertad de decidir la mejor solución para satisfacer lo que se ha pedido en la etapa de diseño. Gracias a las fases anteriores, el programador contará con un análisis y diseño completos del sistema que se debe codificar. Esta actividad implica la creación de programas informáticos aplicando las estructuras de programación de algún paradigma y utilizando un lenguaje de programación apropiado. Al final de este proceso se tendrá un “listado fuente” de los programas que definen el software que se está desarrollando.



3.5 Pruebas del software

Las pruebas del software tienen un doble objetivo; confirmar que la codificación ha sido exitosa y el software no contiene errores, y también que el software haga lo que se supone que debe hacer. Dichas pruebas son un elemento crítico para la garantía de calidad del software y representan una revisión final de las especificaciones del diseño y de la codificación.

Las pruebas pueden realizarse de dos maneras:

- Pruebas de caja negra. Se llevan a cabo sobre la interfaz del software y pretenden demostrar que el software funciona adecuadamente; es decir que las entradas se aceptan de forma adecuada y que se produce también una salida correcta. Estas pruebas no tienen en cuenta la estructura lógica interna del software.
- Pruebas de caja blanca. Estas pruebas se basan en un minucioso examen de los detalles procedimentales. Se comprueban los caminos lógicos del software en base a examinar trozos específicos del programa.

Es común que estos dos tipos de prueba se combinen para validar la interfaz del software y asegurar al mismo tiempo que el funcionamiento interno del software es correcto.

3.6 Mantenimiento

El software producido en la fase de desarrollo debe ser mantenido, ya que sufrirá cambios después de ser entregado al cliente, estos cambios pueden ocurrir debido a lo siguiente:

- Errores encontrados (mantenimiento correctivo).
- Cambios en el entorno externo al que el software debe adaptarse, por ejemplo, la incorporación de un nuevo sistema operativo (mantenimiento adaptativo).
- Que el cliente requiera ampliaciones funcionales o desea incrementar su rendimiento (mantenimiento perfectivo).



3.7 Documentación

La documentación para el usuario debe mostrar la información completa para que comprenda el propósito y la forma correcta de utilizar la aplicación. Además, también debe existir una documentación para el equipo de desarrollo que explica el funcionamiento interno del programa y su codificación.



CAPÍTULO 4. ANÁLISIS



4.1 Análisis de requerimientos

El objetivo de esta etapa es realizar un análisis global del sistema, establecer los requisitos de todos los elementos en él, es una fase de planificación donde se realiza una descripción del entorno software que se quiere obtener y se establecen los recursos necesarios para su desarrollo, así como el coste y un calendario estimado para su desarrollo.

Existen distintas técnicas para el descubrimiento de requerimientos del usuario y del sistema, por ejemplo, la consulta de documentación de las organizaciones, especificaciones de sistemas equivalentes, entrevistas directas o indirectas con los usuarios involucrados, etcétera.

En el caso específico del Centro Escolar se optó por reuniones y entrevistas informales con el personal directivo de la institución, director, subdirector, secretaria escolar, así como con algunos representantes de la Asociación de Padres de Familia, y docentes de grupo para averiguar cuáles son las problemáticas que enfrenta la institución en el proceso de obtener los datos personales de alumnos y sus tutores al inicio del ciclo escolar, y posteriormente su consulta durante y hasta finalizar el ciclo escolar.

A continuación, se enumeran los problemas encontrados en dichas entrevistas y reuniones.

- Falta de claridad en los formularios. Algunos padres de familia señalan que los formularios para la recolección de datos personales son poco claros lo que dificulta su llenado, además de que algunos campos no tienen el espacio suficiente para ser llenados por completo.
- Lentitud en el proceso de matriculación. A casusa de que se deben rellenar formularios de papel para la recolección de los datos personales del alumno y su tutor, el proceso se vuelve demasiado lento, además es importante señalar que debido a la gran cantidad de alumnos que recibe la escuela cada ciclo escolar, la persona encargada de recoger la documentación básica al momento de la inscripción no siempre se percata de que los formularios hayan sido rellenados correctamente, ya que además debe recibir otros documentos importantes para la debida matriculación lo que conlleva un tiempo significativo en el proceso.



- Datos imprecisos y faltantes de información en los formularios. El personal de la dirección escolar señala que la falta de datos en los formularios es muy común, ya que muchas veces el padre de familia o tutor al ignorar algún dato que se le solicita, simplemente prefiere dejar en blanco dicho campo y continuar adelante con el llenado. Otra problemática es que en ocasiones los datos recolectados son inconsistentes debido a que no está claro el formato de los datos que se requieren, por ejemplo, en el caso de la estatura del alumno, algunas veces el padre de familia se confunde y no está seguro de si debe escribir la estatura del alumno en centímetros o en metros, y el caso muy parecido para el campo que pide el peso del alumno. Una problemática similar surge debido a que en la misma hoja del formulario se solicitan datos tanto del alumno como de su tutor, y en ocasiones los tutores llegan a confundirse y escriben los datos que corresponden al tutor en el lugar del alumno o viceversa.
- Búsquedas de información lentas e ineficaces. Los directivos de la institución expresan que el proceso de búsqueda de información personal de algún alumno o su tutor en un momento específico se hace demasiado lenta, ya que es necesario consultar manualmente el engargolado en donde se guarda dicha información.

Derivado de la problemática anterior se propone la construcción e implantación de un sistema web con motor de base de datos que colabore a resolver cada uno de los problemas que se mencionan en los párrafos anteriores. A continuación, se identifican los siguientes requerimientos de usuario.

4.2 Requerimientos de usuario

- Padre de familia o tutor. El sistema proporcionará una interfaz web clara, intuitiva y amigable, visible en dispositivos móviles o fijos, en donde sea posible autenticarse y completar el llenado de datos personales de manera segura, además al finalizar, el sistema emitirá un comprobante de inscripción que servirá para demostrar que el alumno ha sido registrado exitosamente.



- Personal de la dirección escolar. El sistema deberá proporcionar el acceso a los administradores para que estos puedan consultar, actualizar, eliminar, y en su caso agregar información concentrada en él.
- Docentes de grupo. El sistema facilitará el acceso para el registro de calificaciones trimestrales, así como la posibilidad de consultar datos personales de alumnos y tutores del grupo a su cargo.

4.3 Requerimientos del sistema

El sistema de control escolar facilitará la integración de los datos personales de alumnos y sus tutores, y la consulta de toda la información por parte de los administradores del sistema. También posibilitará la captura y consulta de calificaciones trimestrales por los docentes de cada grupo, a su vez éstas podrán ser consultadas también por la dirección escolar.

4.4 Diagramas de casos de uso

Los casos de uso se utilizan durante la obtención de requerimientos y el análisis para representar la funcionalidad del sistema. Los casos de uso se enfocan en el comportamiento del sistema desde un punto de vista externo. Un diagrama de caso de uso describe una función proporcionada por el sistema que produce un resultado visible para un actor. Un actor describe cualquier entidad que interactúa con el sistema. Los actores están fuera de la frontera del sistema, mientras que los casos de uso están dentro de la frontera del sistema. El rectángulo representa los límites del sistema.

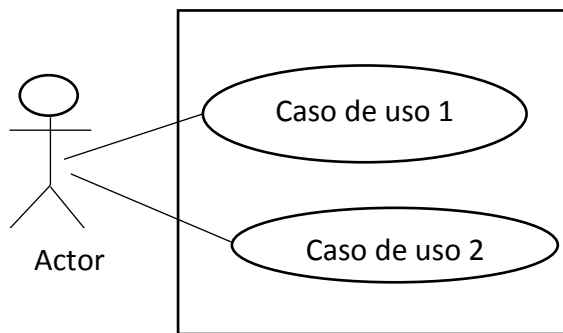


Figura 4.1 Diagrama de caso de uso

En el sistema de control escolar, podemos distinguir básicamente tres actores humanos:

- Padre de familia o tutor
- Docente
- Administrador del sistema

A continuación, se presentan los casos de uso de cada uno de los actores del sistema, es decir, el padre de familia o tutor quien se encarga de ingresar los datos personales del alumno en el sistema, el docente quien es el encargado de capturar las calificaciones de cada alumno a su cargo, y finalmente el administrador del sistema, quien una vez que existan los datos, podrá consultarlos, actualizarlos, o en su caso eliminar o agregar un nuevo registro.

Actor	Casos de uso
<i>Padre de familia o tutor</i>	Ingresar los datos del alumno y su tutor Imprimir comprobante de registro.

Cuadro 4.1 Caso de uso tutor

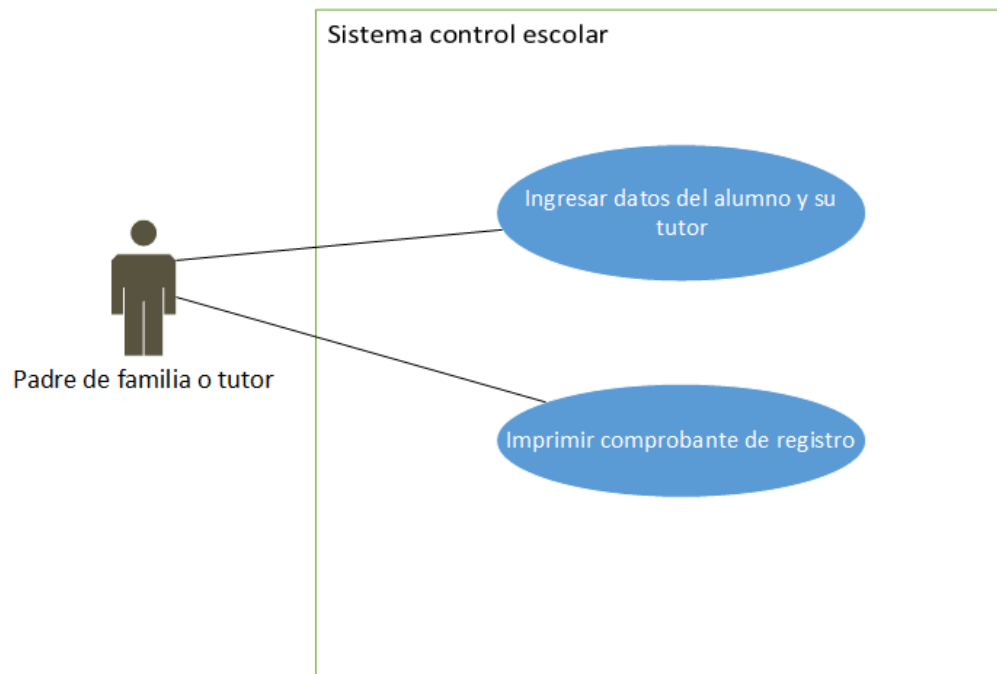


Figura 4.2 Diagrama de casos de uso tutor

Actor	Casos de uso
<i>Docente</i>	Registrar calificaciones trimestrales Actualizar calificaciones Consultar calificaciones Consultar datos de alumnos Consultar datos de tutores

Cuadro 4.2 Casos de uso docente

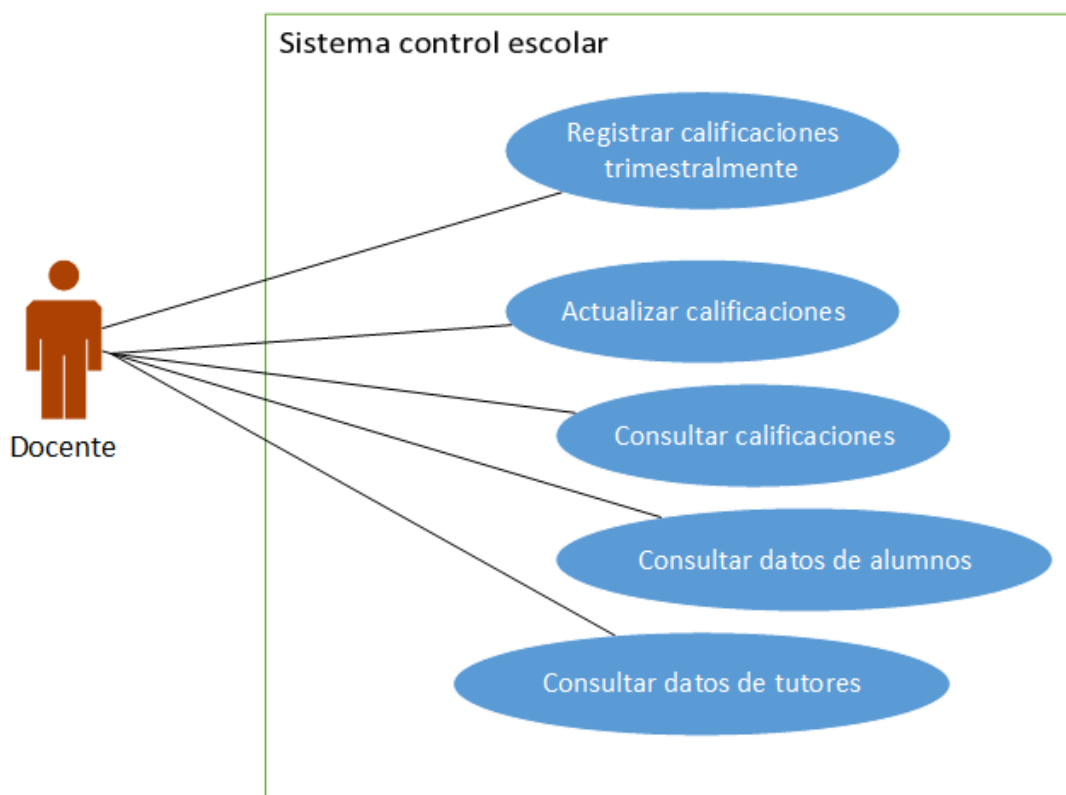


Figura 4.3 Diagrama de casos de uso docente

<i>Actor</i>	Casos de uso
<i>Administrador</i>	Consultar datos de alumnos y tutores Actualizar datos de alumnos y tutores Consultar calificaciones trimestrales por grupo Agregar nuevos alumnos Eliminar alumnos

Cuadro 4.3 Casos de uso administrador

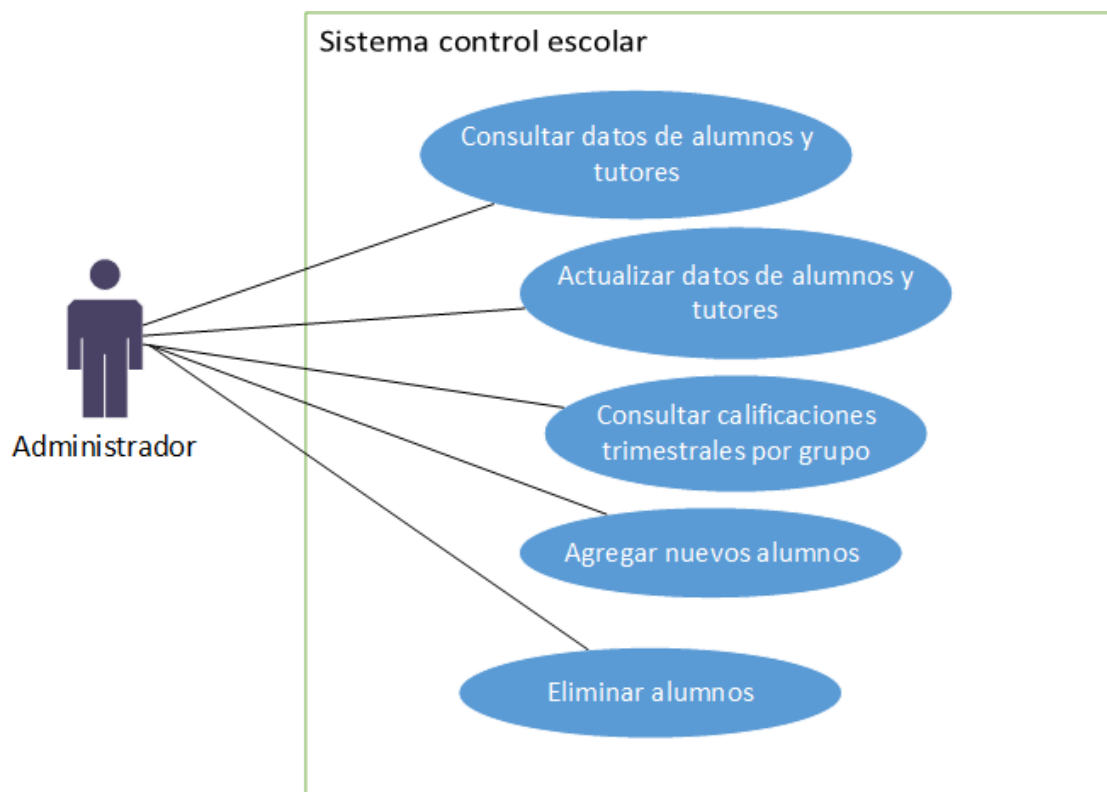


Figura 4.4 Diagrama de casos de uso administrador

4.5 Diagramas de secuencia

Los diagramas de secuencia se utilizan para formalizar el comportamiento del sistema y para visualizar la comunicación entre los objetos que conforman el sistema. Los objetos involucrados en un caso de uso son conocidos como objetos participantes. Un diagrama de secuencia representa las interacciones que suceden entre estos objetos a través del tiempo.

- Las flechas son mensajes que representan la comunicación entre los objetos.
- Las líneas punteadas verticales son llamadas líneas de vida y representan la presencia de un objeto a través del tiempo.
- Los cuadros de activación simbolizan el tiempo que un objeto requiere para realizar su tarea.

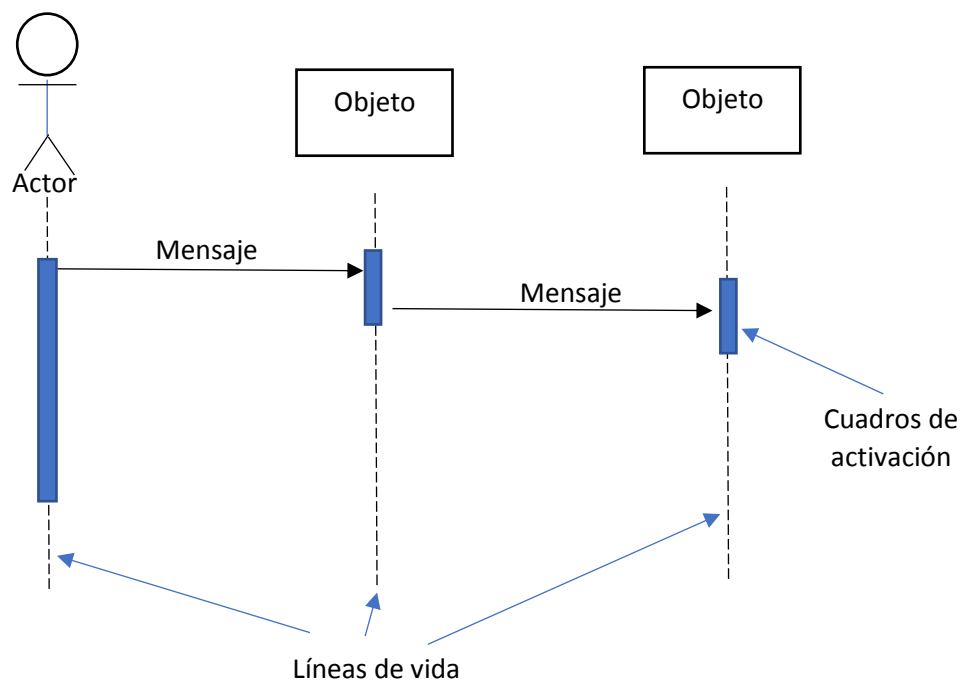


Figura 4.5 Diagrama UML de secuencia de acciones

4.6 Secuencia de acciones para los casos de uso identificados

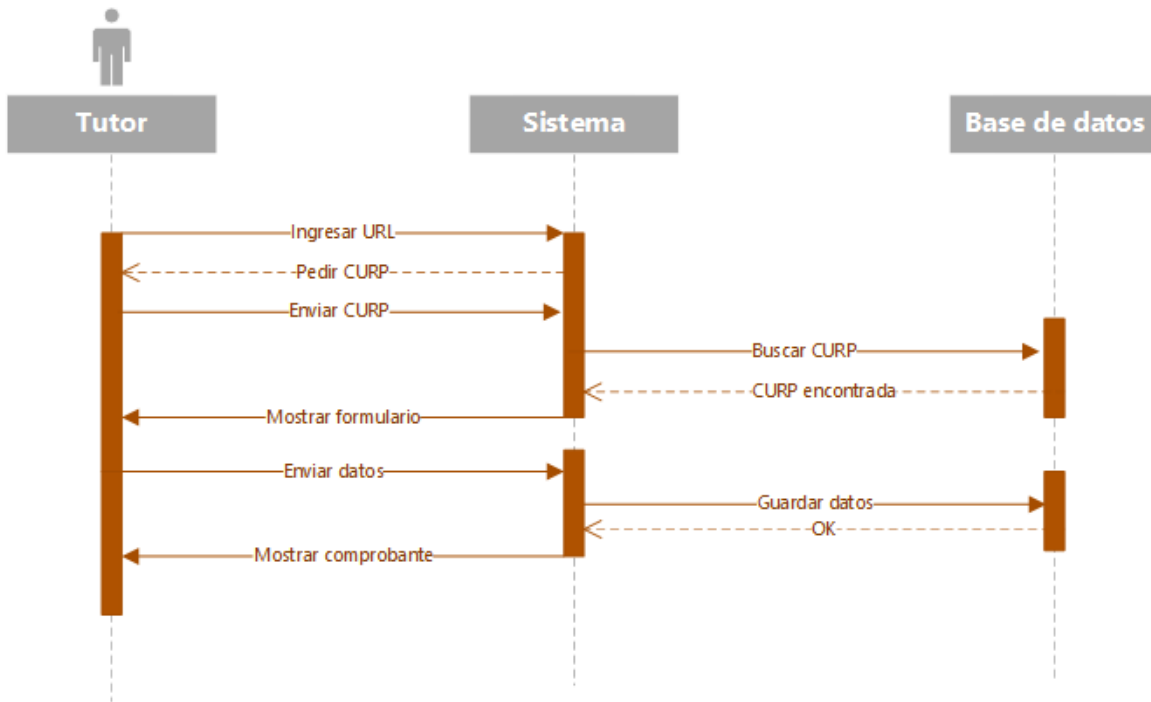


Figura 4.6 Secuencia de acciones para los casos de uso “Ingresar datos del alumno” e “Imprimir comprobante”

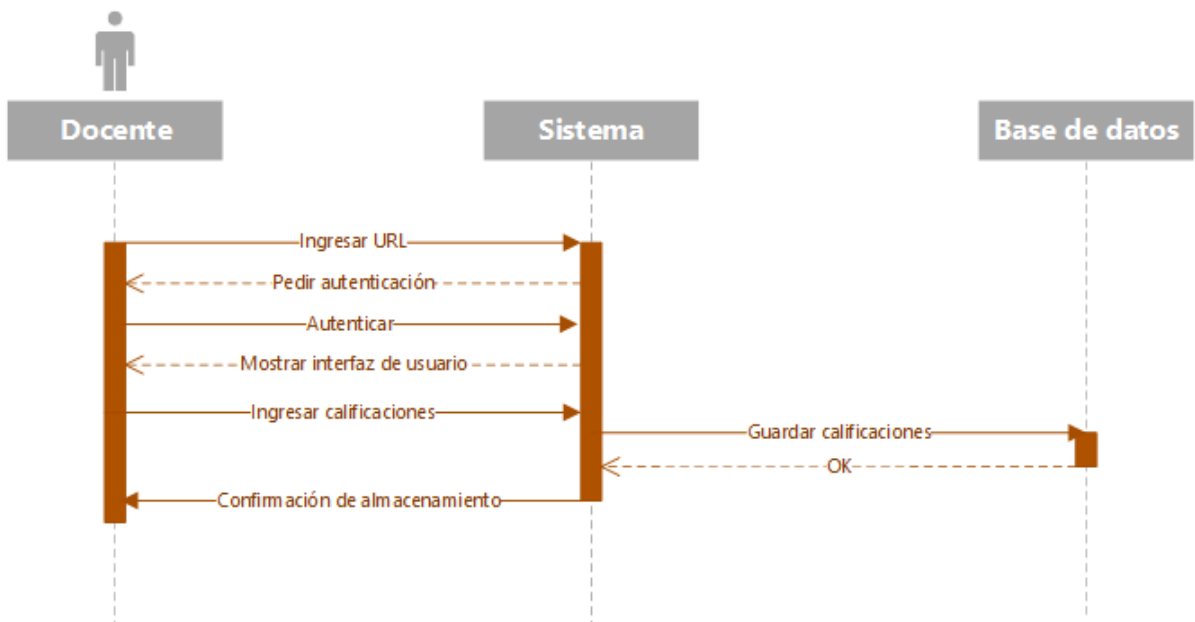


Figura 4.7 Secuencia de acciones para el caso de uso “registrar calificaciones”

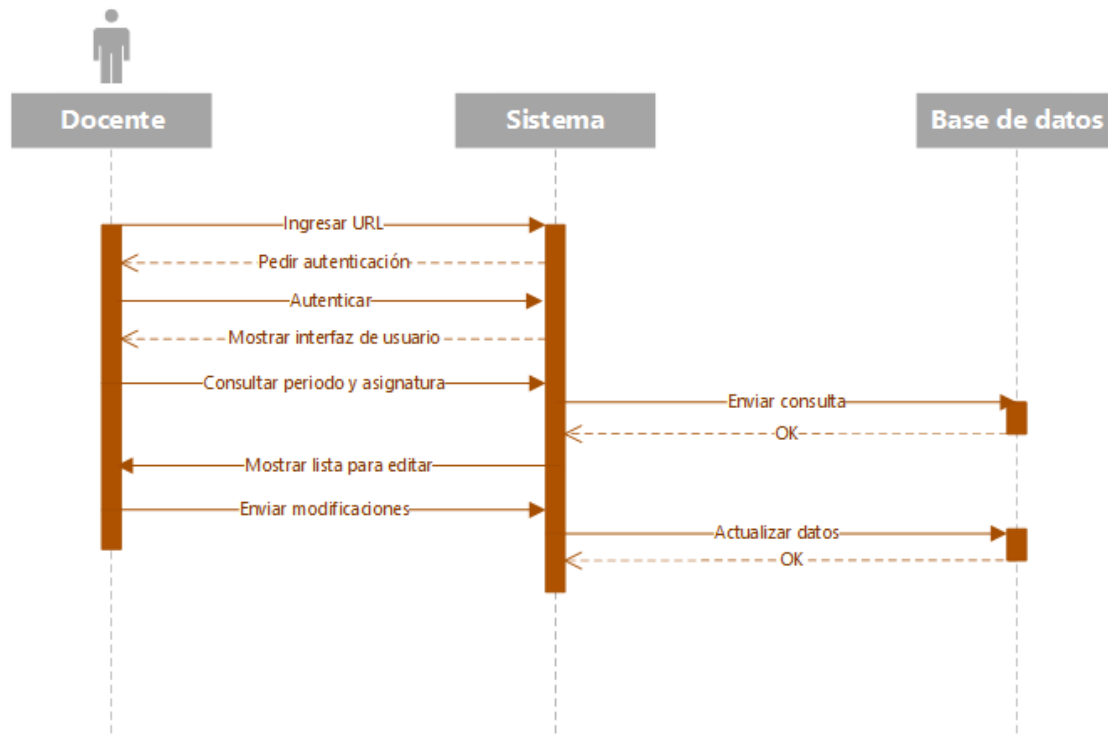


Figura 4.8 Secuencia de acciones para el caso de uso “actualizar calificaciones”

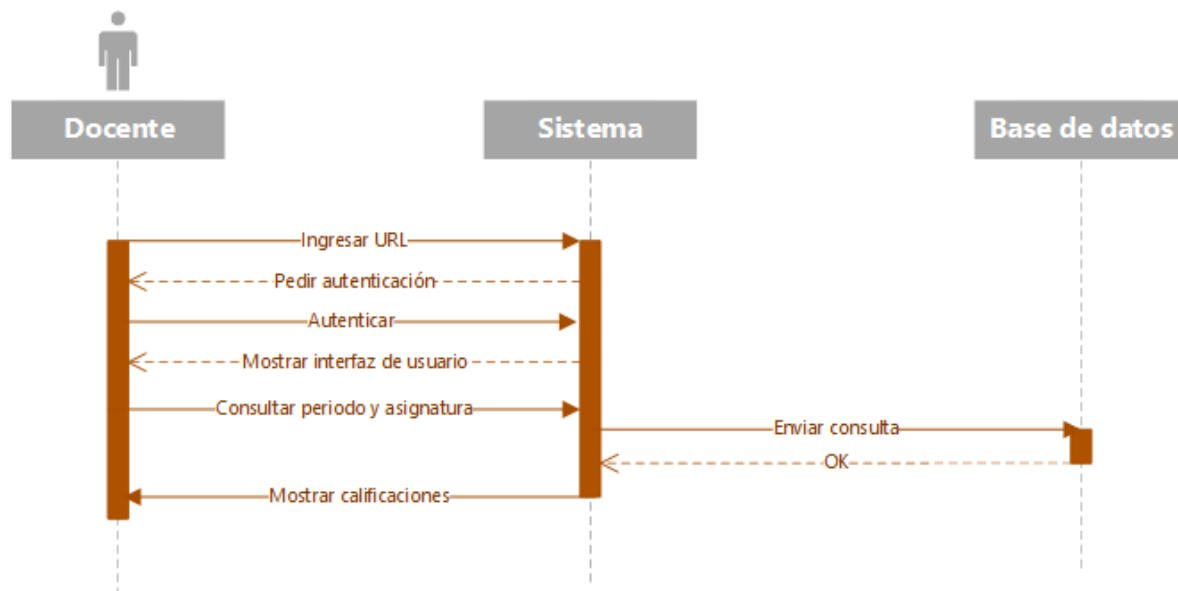


Figura 4.9 Secuencia de acciones para el caso de uso “consultar calificaciones”

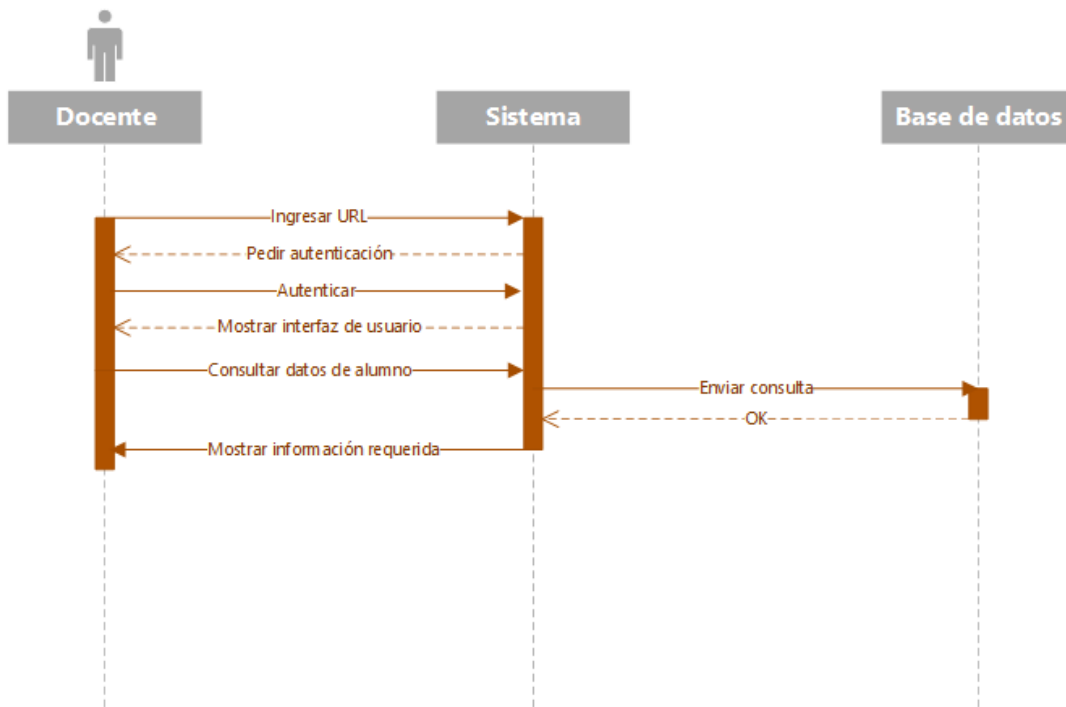


Figura 4.10 Secuencia de acciones para los casos de uso “consultar datos de alumnos”

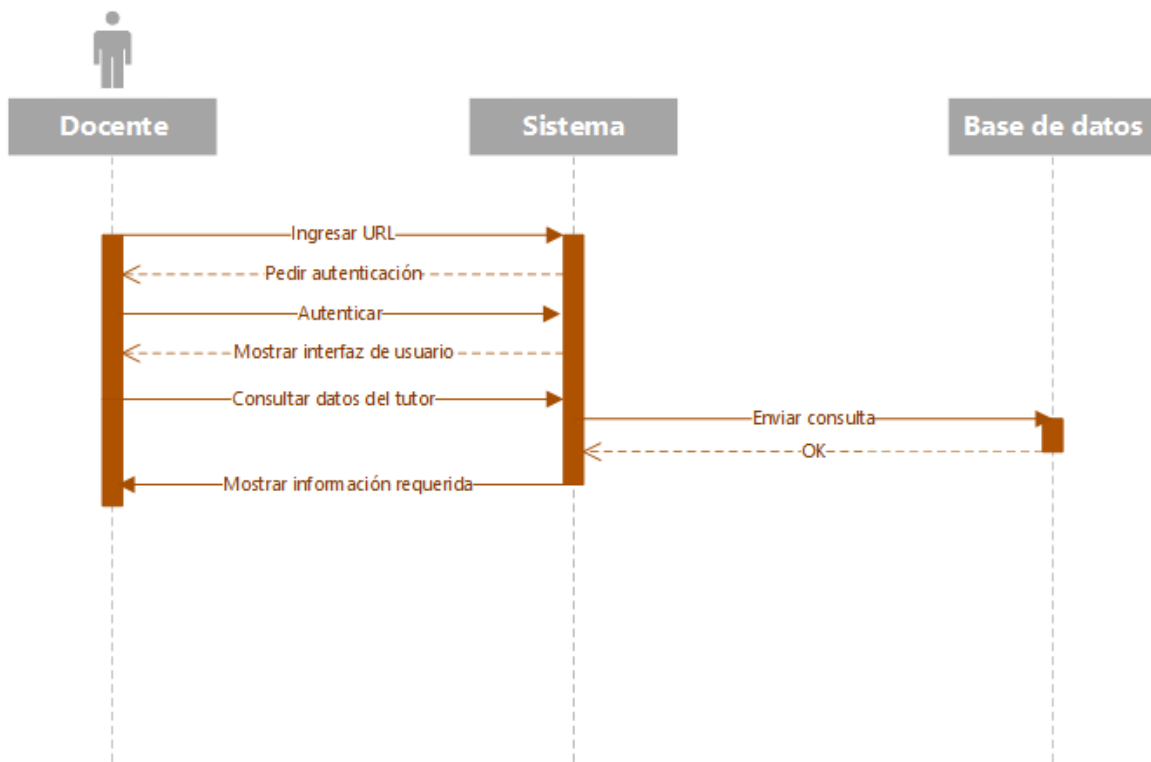


Figura 4.11 Secuencia de acciones para los casos de uso “consultar datos de tutores”

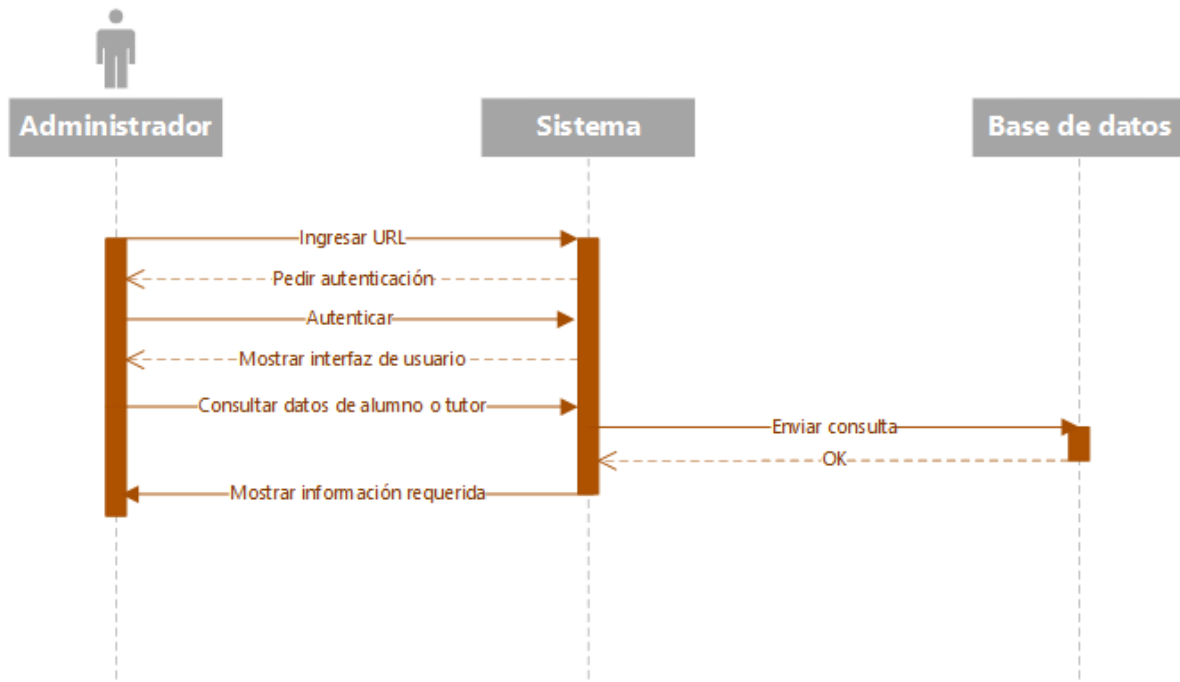


Figura 4.12 Secuencia de acciones para los casos de uso “consultar datos de alumnos y tutores”

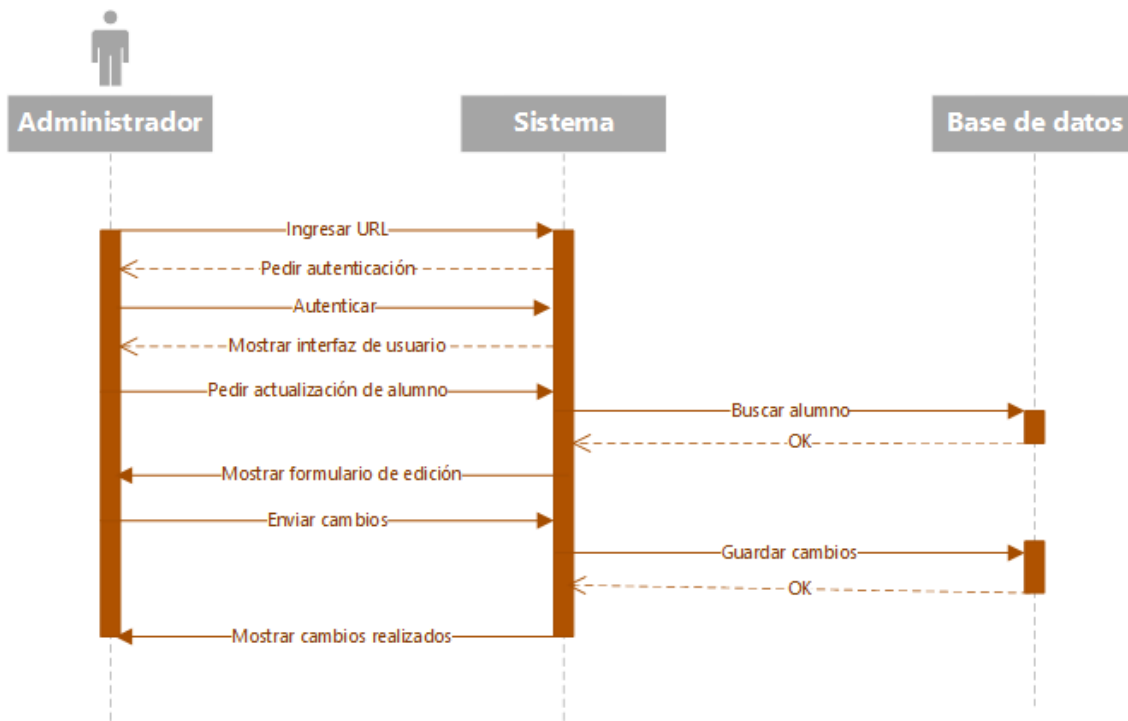


Figura 4.13 Secuencia de acciones para el caso de uso “actualizar datos de alumnos y tutores”

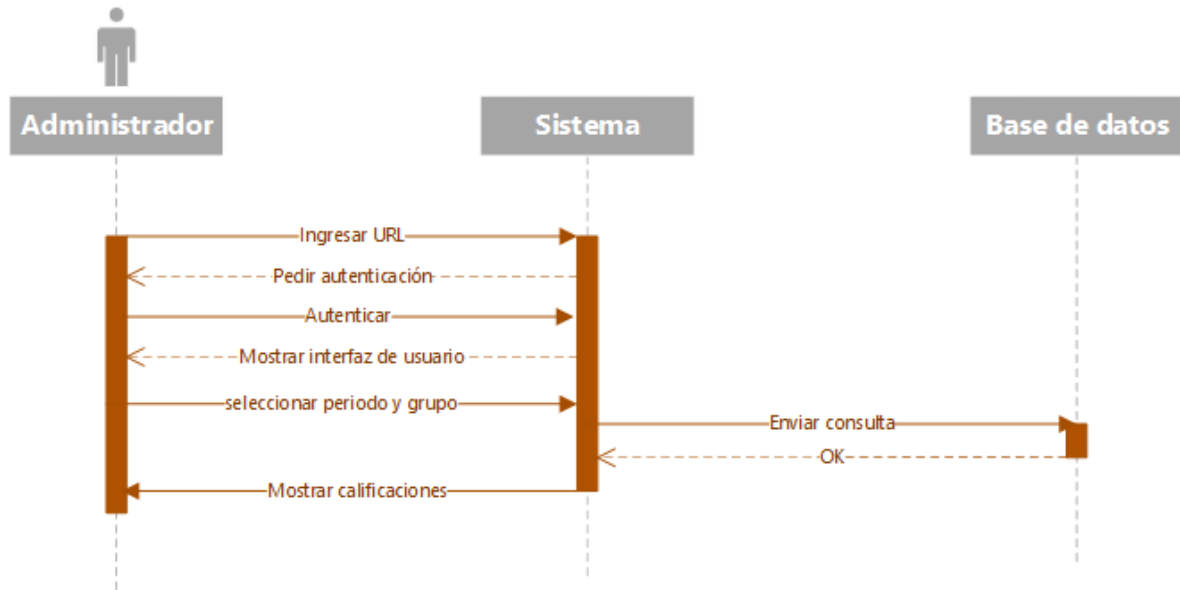


Figura 4.14 Secuencia de acciones para el caso de uso “consultar calificaciones trimestrales por grupo”

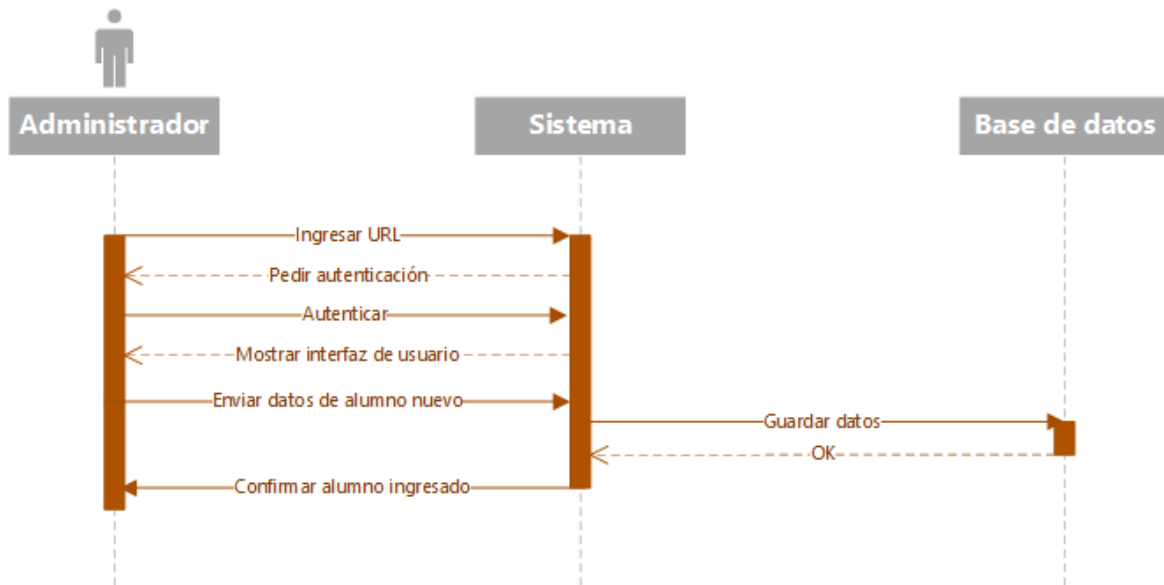


Figura 4.15 Secuencia de acciones para el caso de uso “agregar nuevos alumnos”

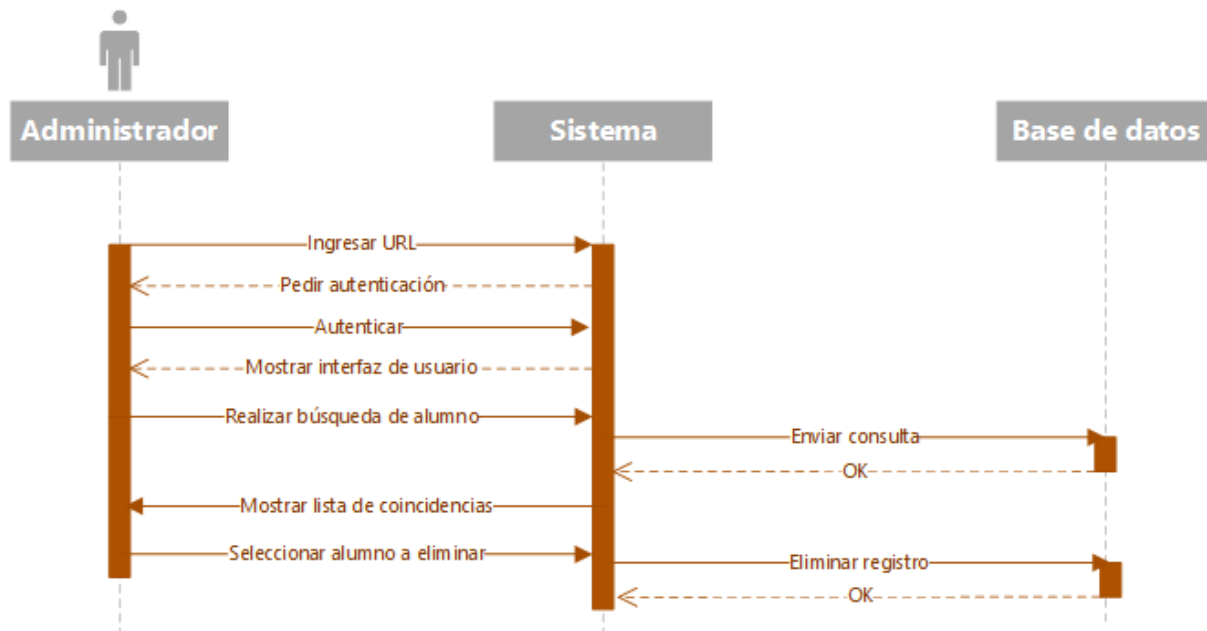


Figura 4.16 Secuencia de acciones para el caso de uso “eliminar alumnos”



4.7 Descripción de escenarios de casos de uso

En las siguientes plantillas se muestran los casos de uso con su respectiva identificación y una breve descripción de lo que hace, actor involucrado, iniciación y flujo estándar de los eventos.

Nombre del caso de uso: Ingresar datos del alumno y su tutor	
Precondiciones: Que la clave CURP del alumno sea válida, es decir que esté digitada correctamente y exista en la base de datos.	
Descripción: El padre de familia ingresa al sistema, digita la clave CURP del alumno y si ésta es aceptada el sistema muestra el formulario para el ingreso de datos del alumno y de su tutor. Al finalizar el proceso de captura el sistema enviará un resumen de los datos ingresados que además servirá como comprobante de inscripción.	
Camino básico	
ACTOR	SISTEMA
1. Ingresar la URL para acceder al sistema de control escolar.	2. Mostrar un cuadro de texto para que el usuario ingrese la clave CURP del alumno.
3. Digitar la clave CURP del alumno.	4. Mostrar formulario para que el usuario pueda ingresar los datos personales del alumno y tutor.
5. Llenar completamente el formulario con los datos del alumno y su tutor y confirmar el envío.	6. Validar cada campo del formulario.
	7. Almacenar los datos.
	8. Enviar comprobante de inscripción.
9. Imprimir comprobante y fin del caso de uso.	
Caminos alternativos:	
Evento 3. Si la clave CURP no es válida, generar un aviso. Evento 6. Si existen campos vacíos en el formulario, generar un aviso.	
Postcondiciones: El alumno habrá quedado registrado exitosamente.	

Cuadro 4.4 Descripción del caso de uso “ingresar datos del alumno y tutor”



Nombre del caso de uso: Registrar calificaciones trimestrales	
Precondiciones: Que el sistema haya habilitado previamente la captura trimestral de calificaciones y que el docente tenga ya listas las calificaciones de los alumnos a su cargo.	
Descripción: El docente ingresa al sistema y se dirige al menú de captura de calificaciones, si aún no está habilitada la captura por el administrador del sistema se muestra un aviso. El docente selecciona una asignatura y después ingresa cada una de las calificaciones de los alumnos del grupo a su cargo.	
Camino básico	
ACTOR	SISTEMA
1. Ingresar al sistema utilizando su nombre de usuario y contraseña.	2. Mostrar la interfaz de usuario.
3. Seleccionar el menú captura de calificaciones.	4. Mostrar lista seleccionable para elegir asignatura.
5. Seleccionar asignatura y hacer clic en el botón capturar.	6. Mostrar lista de alumnos ordenados alfabéticamente y sus respectivos cuadros de texto para la captura de calificaciones.
7. Ingresar calificaciones y hacer clic en el botón para guardar cambios.	8. Almacenar las calificaciones.
9. Salir del sistema y fin del caso de uso.	
Caminos alternativos:	
Evento 3. Si el proceso de captura aún no está habilitado por el administrador, se genera un aviso. Evento 9. El docente puede elegir otra asignatura para continuar ingresando calificaciones.	
Postcondiciones: Las calificaciones de la asignatura seleccionada habrán quedado almacenadas en el sistema.	

Cuadro 4.5 Descripción del caso de uso “registrar calificaciones trimestrales”



Nombre del caso de uso: Actualizar o modificar calificaciones	
Precondiciones: Que el sistema haya habilitado previamente la captura trimestral de calificaciones y que el docente tenga ya listas las calificaciones de los alumnos a su cargo.	
Descripción: El docente ingresa al sistema y se dirige al menú de captura de calificaciones, selecciona una asignatura y después actualiza o modifica las calificaciones necesarias.	
Camino básico	
ACTOR	SISTEMA
1. Ingresar al sistema utilizando su nombre de usuario y contraseña.	2. Mostrar la interfaz de usuario.
3. Seleccionar el menú captura de calificaciones.	4. Mostrar lista seleccionable para elegir asignatura.
5. Seleccionar trimestre, asignatura y hacer clic en el botón capturar / modificar.	6. Mostrar lista de alumnos ordenados alfabéticamente y sus respectivos cuadros de texto para la captura o modificación de las calificaciones.
7. Actualizar calificaciones y hacer clic en el botón para guardar cambios.	8. Guardas los cambios realizados a las calificaciones.
9. Salir del sistema y fin del caso de uso.	
Caminos alternativos:	
Evento 3. Si el proceso de captura aún no está habilitado por el administrador, se genera un aviso. Evento 9. El docente puede elegir otra asignatura para continuar actualizando o modificando calificaciones.	
Postcondiciones: Los cambios realizados a las calificaciones habrán quedado guardadas en el sistema.	

Cuadro 4.6 Descripción del caso de uso “actualizar o modificar calificaciones”



Nombre del caso de uso: Consultar calificaciones	
Precondiciones: Que existan calificaciones del trimestre y asignatura que se desea consultar.	
Descripción: El docente ingresa al sistema y se dirige al menú de consulta de calificaciones, elige un periodo trimestral y una asignatura, después hace clic en el botón consultar para visualizar las calificaciones del grupo a su cargo.	
Camino básico	
ACTOR	SISTEMA
1. Ingresar al sistema utilizando su nombre de usuario y contraseña.	2. Mostrar la interfaz de usuario.
3. Seleccionar el menú para la consulta de calificaciones.	4. Mostrar controles para seleccionar el trimestre y la asignatura.
5. Seleccionar trimestre y asignatura, después hacer clic en el botón consultar.	6. Mostrar lista de alumnos ordenados alfabéticamente con sus respectivas calificaciones.
7. Salir del sistema y fin del caso de uso.	
Caminos alternativos:	
Evento 6. El docente puede elegir otro trimestre y asignatura para continuar consultando calificaciones de su grupo.	
Postcondiciones: No aplica.	

Cuadro 4.7 Descripción del caso de uso “consultar calificaciones”



Nombre del caso de uso: Consultar datos del alumno	
Precondiciones: Que los datos del alumno a consultar existan en la base de datos.	
Descripción: El docente ingresa al sistema y se dirige al menú de consulta de grupo, se muestra una lista de alumnos en donde es posible hacer clic en un botón para visualizar el expediente completo del alumno a su cargo.	
Camino básico	
ACTOR	SISTEMA
1. Ingresar al sistema utilizando su nombre de usuario y contraseña.	2. Mostrar la interfaz de usuario.
3. Seleccionar el menú para la consulta de grupo.	4. Mostrar lista de alumnos.
5. Seleccionar un alumno haciendo clic en el botón datos del alumno.	6. Mostrar página con los datos completos del alumno.
7. Salir del sistema y fin del caso de uso.	
Caminos alternativos:	
Evento 6. El docente puede regresar y elegir otro alumno de la lista para consultar sus datos.	
Postcondiciones: No aplica.	

Cuadro 4.8 Descripción del caso de uso “consultar datos del alumno”



Nombre del caso de uso: Consultar datos del tutor	
Precondiciones: Que los datos del tutor a consultar existan en la base de datos.	
Descripción: El docente ingresa al sistema y se dirige al menú de consulta de grupo, se muestra una lista de alumnos en donde es posible hacer clic en un botón para visualizar los datos completos del tutor.	
Camino básico	
ACTOR	SISTEMA
1. Ingresar al sistema utilizando su nombre de usuario y contraseña.	2. Mostrar la interfaz de usuario.
3. Seleccionar el menú para la consulta de grupo.	4. Mostrar lista de alumnos.
5. Seleccionar un tutor haciendo clic en el botón datos del tutor.	6. Mostrar página con los datos completos del tutor.
7. Salir del sistema y fin del caso de uso.	
Caminos alternativos:	
Evento 6. El docente puede regresar y elegir otro tutor de la lista para consultar sus datos.	
Postcondiciones: No aplica.	

Cuadro 4.9 Descripción del caso de uso “consultar datos del tutor”



Nombre del caso de uso: Consultar datos de alumnos y tutores	
Precondiciones: Que los datos del alumno y su tutor a consultar existan en la base de datos.	
Descripción: El administrador ingresa al sistema y se dirige al menú de consulta de grupos, selecciona el grado, el grupo y se muestra una lista con el nombre de los alumnos del grupo seleccionado. El administrador puede elegir visualizar los datos completos de un alumno o su tutor haciendo clic en el botón respectivo.	
Camino básico	
ACTOR	SISTEMA
1. Ingresar al sistema utilizando su nombre de usuario y contraseña.	2. Mostrar la interfaz de usuario.
3. Seleccionar el menú para la consulta de grupos.	4. Mostrar lista de alumnos del grupo seleccionado.
5. Seleccionar de la lista un alumno o tutor haciendo clic en el botón correspondiente.	6. Mostrar página con los datos completos del alumno o su tutor.
7. Salir del sistema y fin del caso de uso.	
Caminos alternativos:	
Evento 4. El administrador puede elegir visualizar un grupo distinto seleccionando el grado y grupo deseado.	
Evento 6. El administrador puede regresar y elegir otro alumno o tutor de la lista para consultar sus datos.	
Postcondiciones: No aplica.	

Cuadro 4.10 Descripción del caso de uso “consultar datos de alumnos y tutores”



Nombre del caso de uso: Actualizar datos de alumnos y tutores	
Precondiciones: Que los datos del alumno y su tutor a consultar existan en la base de datos.	
Descripción: El administrador ingresa al sistema y se dirige al menú de consulta de grupos, selecciona el grado, el grupo y se muestra una lista con el nombre de los alumnos del grupo seleccionado. El administrador puede elegir actualizar los datos completos de un alumno o su tutor haciendo clic en el botón respectivo.	
Camino básico	
ACTOR	SISTEMA
1. Ingresar al sistema utilizando su nombre de usuario y contraseña.	2. Mostrar la interfaz de usuario.
3. Seleccionar el menú para la consulta de grupos.	4. Mostrar lista de alumnos del grupo seleccionado.
5. Seleccionar de la lista un alumno o tutor haciendo clic en el botón de edición correspondiente.	6. Mostrar formulario de edición con los datos completos del alumno o tutor.
7. Modificar los campos deseados y hacer clic en el botón actualizar datos.	8. Actualizar la información en la base de datos.
9. Salir del sistema y fin del caso de uso.	
Caminos alternativos:	
Evento 4. El administrador puede elegir visualizar un grupo distinto seleccionando el grado y grupo deseado.	
Evento 6. El administrador puede regresar y elegir otro alumno o tutor de la lista para modificar sus datos.	
Postcondiciones: Las modificaciones en los datos de alumnos o tutores habrán sido almacenadas en la base de datos.	

Cuadro 4.11 Descripción del caso de uso “actualizar datos de alumnos y tutores”



Nombre del caso de uso: Consultar calificaciones por grupo	
Precondiciones: Que existan calificaciones del trimestre y asignatura que se desea consultar.	
Descripción: El administrador ingresa al sistema y se dirige al menú de consulta de calificaciones, elige un periodo trimestral y el grupo que desea consultar, enseguida hace clic en el botón consultar y visualiza las calificaciones de todas las asignaturas en el trimestre elegido.	
Camino básico	
ACTOR	SISTEMA
1. Ingresar al sistema utilizando su nombre de usuario y contraseña.	2. Mostrar la interfaz de usuario.
3. Seleccionar el menú para la consulta de calificaciones.	4. Mostrar controles para seleccionar el trimestre y grupo.
5. Seleccionar trimestre y grupo, después hacer clic en el botón consultar.	6. Mostrar lista de alumnos ordenados alfabéticamente con las calificaciones correspondientes de todas las asignaturas.
7. Salir del sistema y fin del caso de uso.	
Caminos alternativos:	
Evento 6. El administrador puede elegir un trimestre y grupo distintos y continuar consultando calificaciones.	
Postcondiciones: No aplica.	

Cuadro 4.12 Descripción del caso de uso “consultar calificaciones por grupo”



Nombre del caso de uso: Agregar nuevos alumnos	
Precondiciones: Que la dirección escolar haya aprobado previamente el alta del nuevo alumno y que además tenga a la mano los datos básicos del alumno que son el nombre completo y su clave CURP.	
Descripción: El administrador entra al sistema y se dirige al menú nuevo ingreso, en donde captura en los campos correspondientes el nombre completo del alumno, así como su clave CURP. Finalmente, el administrador guardará los datos del nuevo alumno haciendo clic en el botón correspondiente.	
Camino básico	
ACTOR	SISTEMA
1. Ingresar al sistema utilizando su nombre de usuario y contraseña.	2. Mostrar la interfaz de usuario.
3. Seleccionar el menú nuevo ingreso.	4. Mostrar formulario.
5. Capturar el nombre completo del alumno, así como su clave CURP, después hacer clic en el botón ingresar.	6. Guardar datos del nuevo alumno.
7. Salir del sistema y fin del caso de uso.	
Caminos alternativos:	
Evento 5. Si los datos del alumno ya existen, generar un aviso.	
Postcondiciones: Los datos básicos del nuevo alumno habrán sido almacenados.	

Cuadro 4.13 Descripción del caso de uso “agregar nuevos alumnos”



Nombre del caso de uso: Eliminar alumnos	
Precondiciones: Que el alumno exista en la base de datos.	
Descripción: El administrador entra al sistema y se dirige al menú de búsqueda de alumnos, escribe el nombre del alumno y se muestra una lista de coincidencias. El administrador selecciona un alumno de la lista y hace clic en el botón para eliminar al alumno.	
Camino básico	
ACTOR	SISTEMA
1. Ingresar al sistema utilizando su nombre de usuario y contraseña.	2. Mostrar la interfaz de usuario.
3. Seleccionar el menú búsqueda.	4. Mostrar campo de búsqueda.
5. Escribir parte del nombre o el nombre completo del alumno que se desea eliminar.	6. Realizar búsqueda de coincidencias y mostrar los resultados en forma de lista.
7. Elegir el alumno y hacer clic en el botón para eliminarlo del sistema.	8. Pedir confirmación para proceder a eliminar el registro.
9. Confirmar eliminación definitiva.	10. Eliminar el registro del sistema.
11. Salir del sistema y fin del caso de uso.	
Caminos alternativos:	
Evento 5. Realizar nueva búsqueda escribiendo un nombre diferente. Evento 9. Cancelar la eliminación.	
Postcondiciones: El alumno habrá sido eliminado del sistema.	

Cuadro 4.14 Descripción del caso de uso “eliminar alumnos”



CAPÍTULO 5. DISEÑO





5.1 Arquitectura cliente – servidor

La arquitectura cliente – servidor es un modelo de aplicación distribuida en el que las tareas se reparten entre los proveedores de recursos o servicios, llamados servidores, y los demandantes, llamados clientes. Un cliente realiza peticiones a otro programa, el servidor, y éste a su vez le envía una respuesta.

En esta arquitectura la capacidad de proceso está repartida entre los clientes y los servidores, esta separación es de tipo lógico, donde el servidor no se ejecuta necesariamente sobre una sola máquina. Los tipos específicos de servidores incluyen los servidores web, servidores de archivo, servidores de correo, etcétera. Aunque los objetivos de cada servidor sean distintos unos de otros, la arquitectura básica es la misma.

En los sistemas multicapa el servidor se divide en distintos programas que pueden ser ejecutados por diferentes computadoras acrecentando así el grado de distribución del sistema. La arquitectura cliente - servidor a diferencia de la arquitectura monolítica existe una distribución a nivel físico y también a nivel lógico.

La distribución a nivel físico plantea distribuir toda infraestructura de hardware que ejecuta los procesos en niveles, donde cada elemento con una función distinta representa una capa aparte. Por otro lado, a nivel lógico las aplicaciones web siguen una arquitectura de capas dividiendo la forma en cómo se organiza el código de la aplicación sin tener en cuenta su distribución física. Tomando en cuenta esto último, es posible tener una aplicación web completa ejecutándose en una sola computadora o en varias, sin embargo, el concepto de arquitectura lógica en capas esencialmente es el mismo.

Una aplicación diseñada según el modelo de tres capas se divide en presentación, lógica de negocio y persistencia o almacenamiento de datos.

- Capa de presentación. La capa de presentación es la encargada de capturar la interacción entre el usuario y el sistema, contiene todos los elementos con los que el usuario puede interactuar a través de una interfaz gráfica visible a través de un navegador web dentro de algún dispositivo de acceso como una computadora de escritorio, portátil, smartphone o tablet. Los componentes dentro de esta capa solicitan y reciben servicios de la misma capa o desde la capa de lógica de negocio.



- Capa de lógica de negocio o capa intermedia. Aquí se modela el comportamiento del sistema estableciendo reglas y realizando los cálculos y procesos pertinentes con los datos. En esta capa se reciben las peticiones del usuario y se muestran también los resultados después de ser procesados. Básicamente la capa intermedia es la encargada de mover y procesar los datos entre las capas de presentación y de datos. Normalmente la capa de negocio se ejecuta en un equipo de cómputo llamado servidor de aplicaciones.
- Capa de persistencia o datos. Es la encargada del acceso a los datos. Está compuesta por el esquema de datos propio de cada aplicación, así como un sistema gestor de base de datos (SGBD) que se ejecutan en un servidor de base de datos los cuales se encargan de los procesos de solicitudes de registro, almacenamiento, modificación o consulta desde la capa de lógica de negocio.

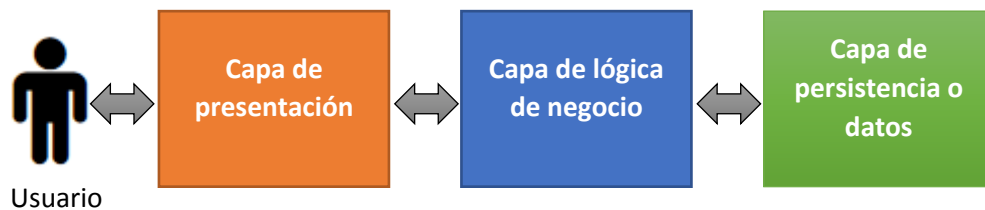


Figura 5.1 El modelo de 3 capas

5.2 Diseño de la interfaz de usuario

Las interfaces de usuario basadas en web, tienen como fundamento el soporte proporcionado por el lenguaje HTML que es acrónimo de HyperText Markup Language, o lenguaje marcado de hipertexto en español.

El lenguaje HTML. HTML (HyperText Markup Language) un lenguaje de marcado de hiper texto para la descripción de contenido en una página web.

El protocolo HTTP. El protocolo HTTP (HyperText Transfer Protocol) es un protocolo de transferencia de hipertexto que se encarga de las transacciones de información a través de



Internet. El objetivo de este protocolo es permitir la transferencia de archivos en formato HTML entre un navegador y un servidor web a través de una cadena de caracteres conocida como URL (Uniform Resource Locator) localizador uniforme de recursos.

Debido a que se desarrollará una aplicación web, la capa de presentación de la arquitectura utilizará un navegador web como Chrome, Explorer, Firefox o Safari, para interactuar con el usuario. Con base en los requerimientos de usuario se determinó diseñar una interfaz basada en formularios, que permita el ingreso y lectura de datos fácilmente. A continuación, se presentan algunos prototipos en baja fidelidad de la interfaz de usuario propuesta. Estos prototipos se discutieron previamente con los usuarios finales de la aplicación para decidir sus características, organización y apariencia.

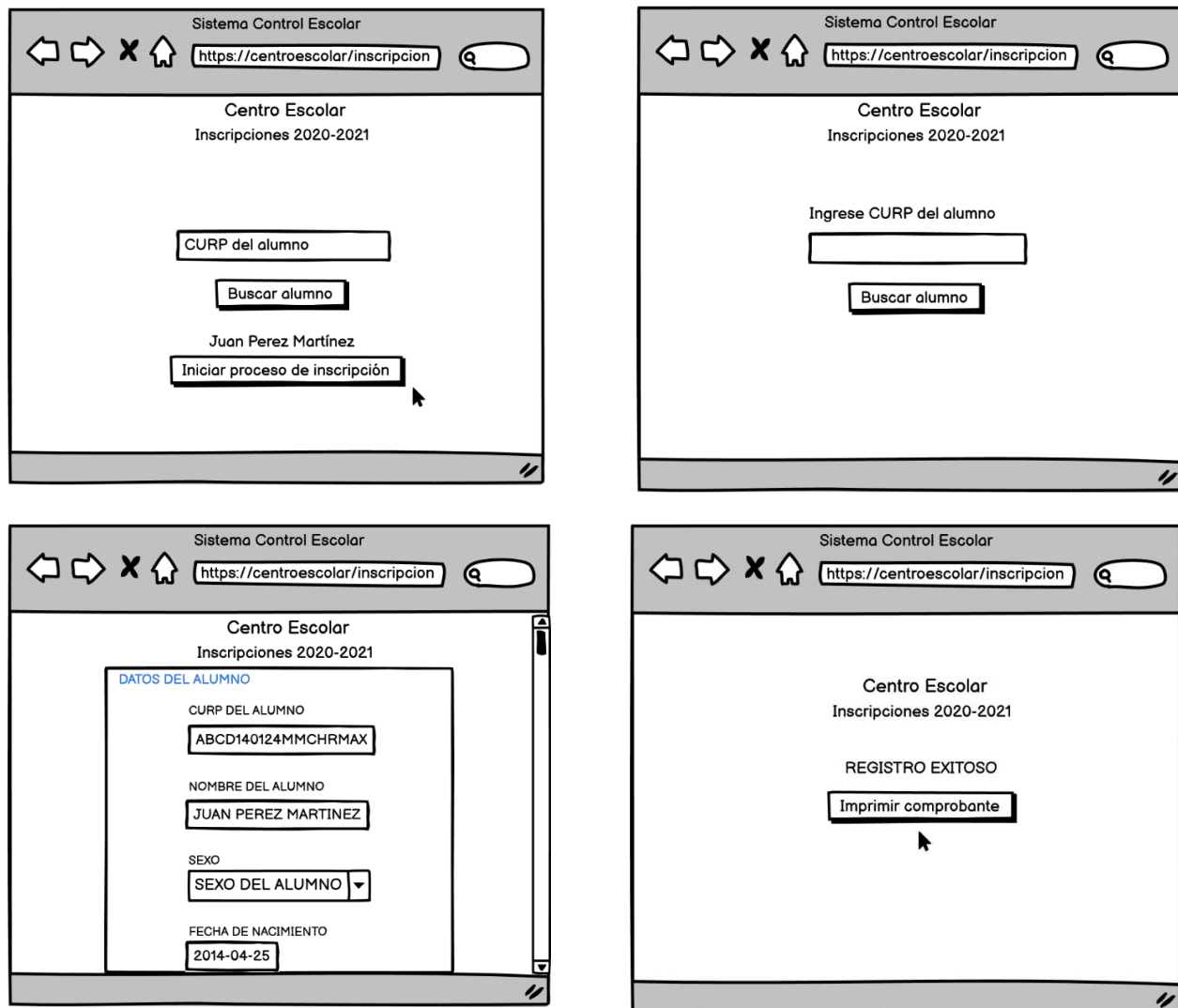


Figura 5.2 Interfaces de usuario para el tutor

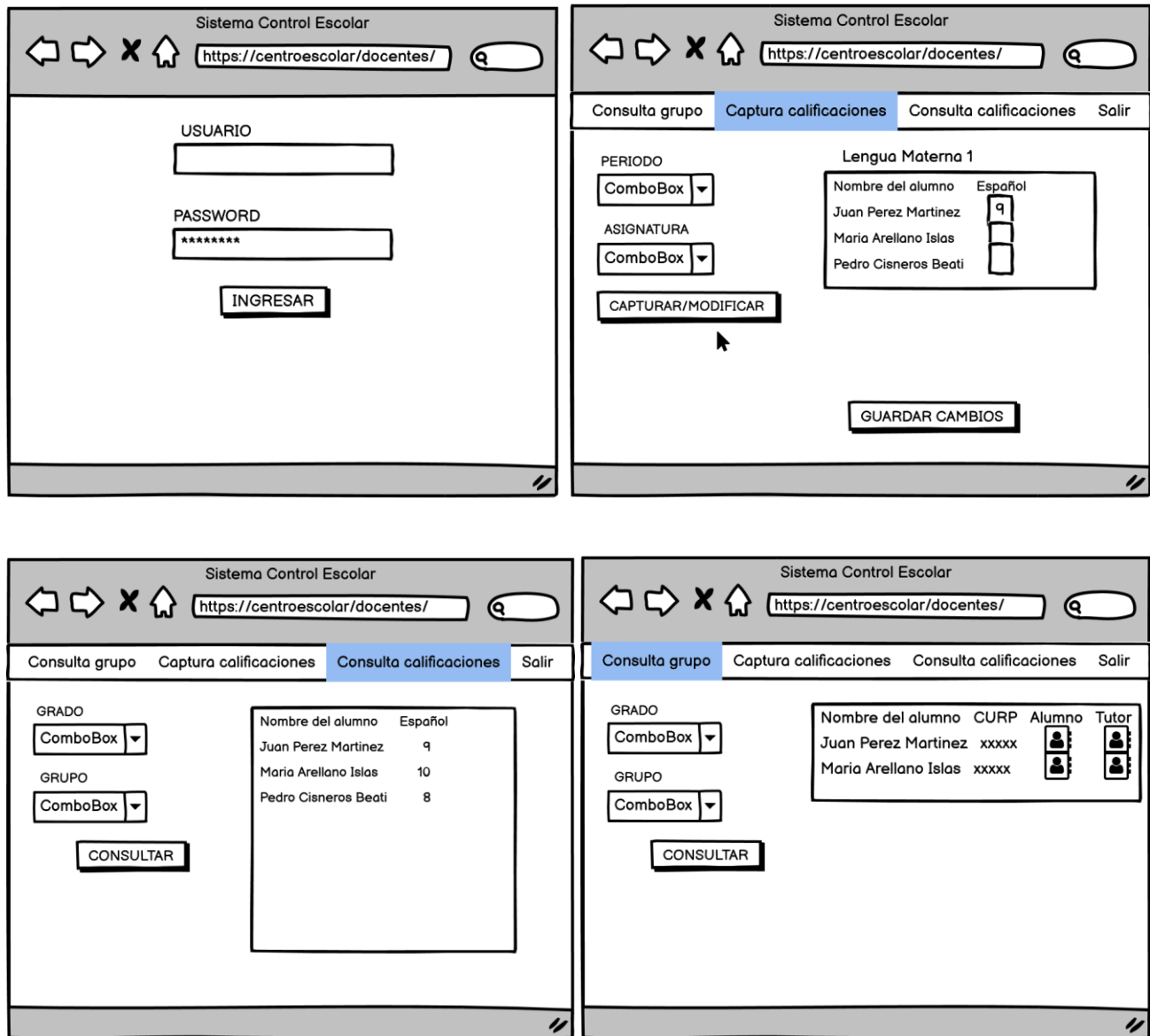


Figura 5.3 Interfaces de usuario para el docente

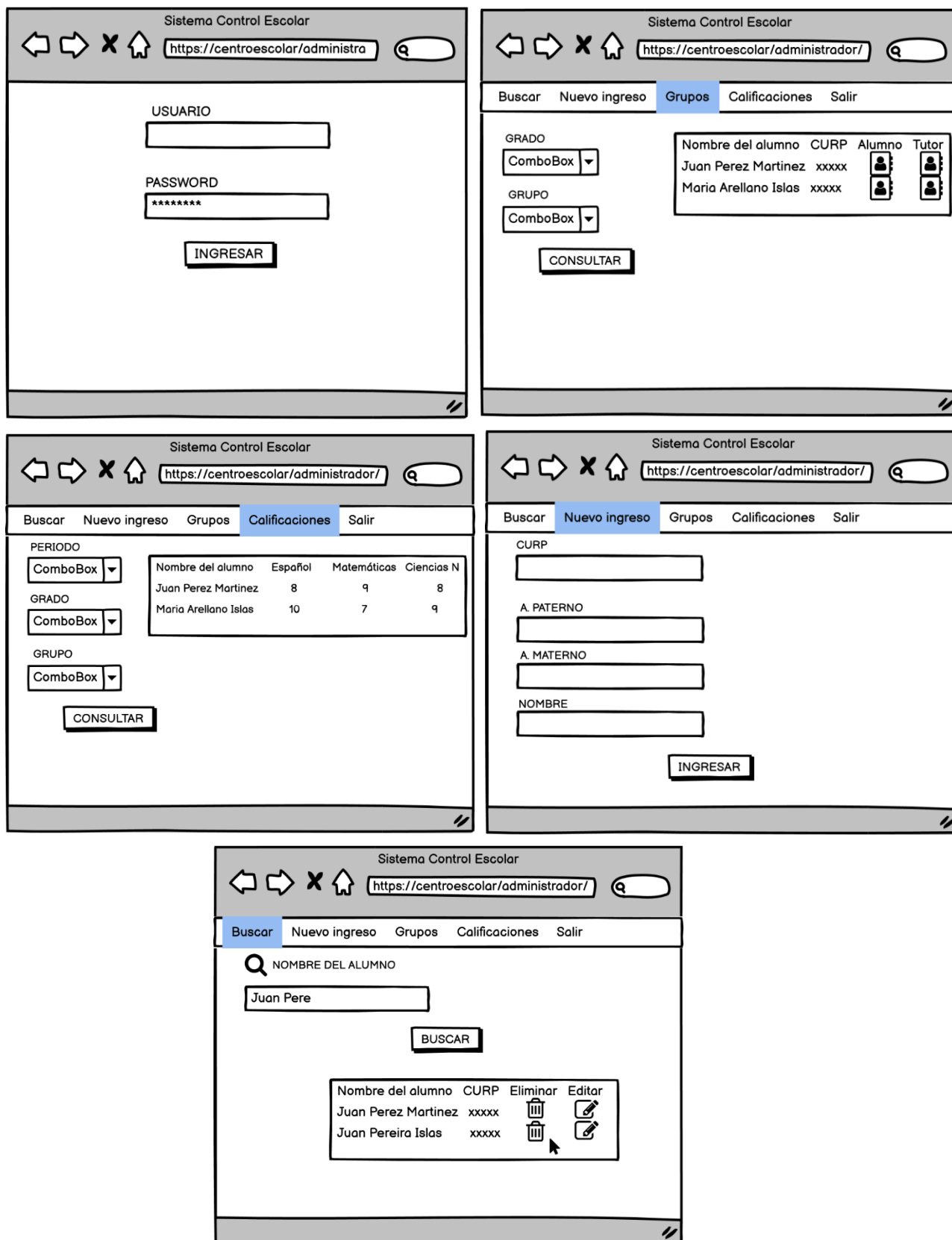


Figura 5.4 Interfaces de usuario para el administrador



5.3 Diseño conceptual de la base de datos

El diseño conceptual toma como base las especificaciones de usuario para obtener una representación del mundo real llamado modelo conceptual. En este modelo se describen las entidades, sus propiedades o atributos y las relaciones existentes entre ellas.

Modelo conceptual entidad – relación

El modelo entidad – relación (E-R) es una técnica que permite representar de forma esquemática un diagrama denominado diagrama ER, de un sistema de información siguiendo una metodología gráfica basada en reglas, símbolos y métodos para diseñar bases de datos. Con esta metodología se pretende obtener un modelo abstracto que represente la información derivada del mundo real de una manera gráfica. Los tres conceptos más importantes de entender son: Entidad, atributo y relación.

Entidad. Es una representación de un objeto del mundo real, y que puede diferenciarse de otros objetos, por ejemplo, algunas entidades válidas podrían ser clientes, vendedores, productos, alumnos, libros, etcétera. Además, las entidades pueden clasificarse en concretas y abstractas, son abstractas cuando representan objetos tangibles del mundo real, por ejemplo, personas, clientes o libros. Por otro lado, las entidades abstractas representan objetos intangibles como son: pagos, asignaturas, inscripciones, etcétera.

Además de que las entidades pueden clasificarse en concretas o abstractas, pueden ser de dos tipos: Fuertes o débiles.

Entidad fuerte. Son entidades normales que tienen existencia independiente, es decir su existencia no depende de otras entidades. Su representación gráfica es un rectángulo.

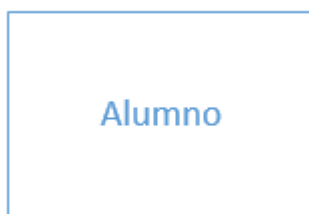


Figura 5.5 Representación de una entidad fuerte

Entidad débil. Su existencia depende de otras, por ejemplo, una entidad llamada calificación depende de una entidad fuerte llamada alumno.



Figura 5.6 Representación de una entidad débil

Relación. Una relación es una asociación que vincula las entidades del modelo. En el modelo ER una relación se representa con un rombo unido por líneas a cada una de las entidades asociadas, y un nombre que la define unívocamente en el diagrama, por ejemplo:



Figura 5.7 Representación de una relación de entidades

Por lo general se modelan relaciones binarias entre dos entidades, por ejemplo, en las entidades anteriores se puede observar que un tutor pertenece a uno y solo un alumno, y que un tutor puede tener de 1 a M alumnos.

Atributo. Los atributos especifican las propiedades de las entidades y las relaciones, su valor debe ser único para cada entidad. Se representa con un círculo y dentro de él, se escribe el nombre del atributo.

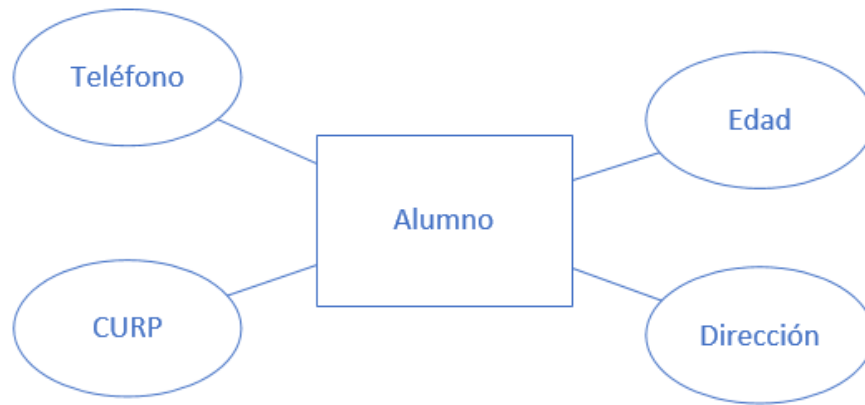


Figura 5.8 Representación de atributos de una entidad

Cardinalidad. Establece el número de relaciones que pueden aparecer en una entidad.

Pueden ser de dos tipos:

- Cardinalidad mínima. Establece el número mínimo de relaciones en las que aparecerá cada ejemplar de la entidad.
- Cardinalidad máxima. Señala el número máximo de relaciones en las que puede aparecer cada ejemplar de la entidad.

A continuación, se muestra el modelo conceptual con las entidades, relaciones y atributos sintetizados del sistema de control escolar.

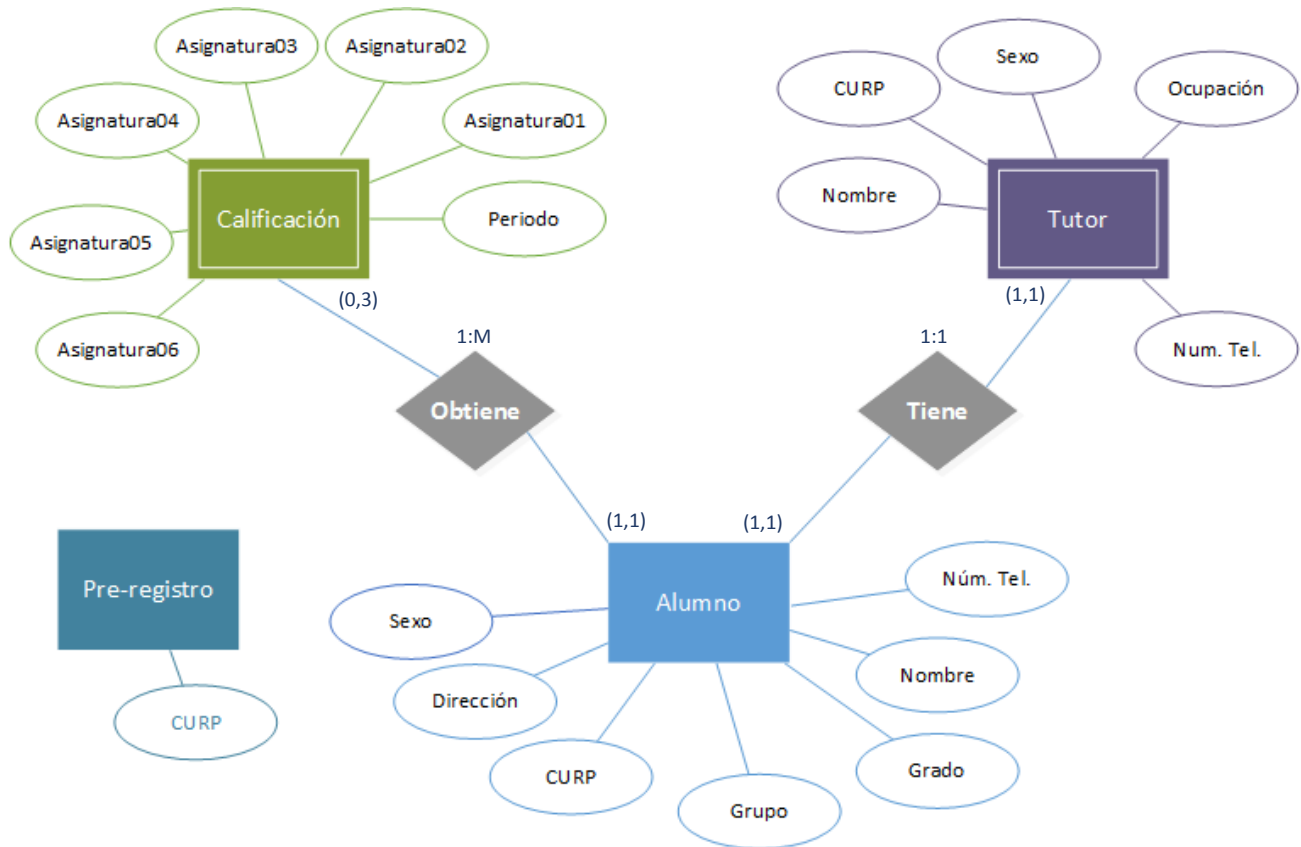


Figura 5.9 Modelo conceptual para el sistema de control escolar

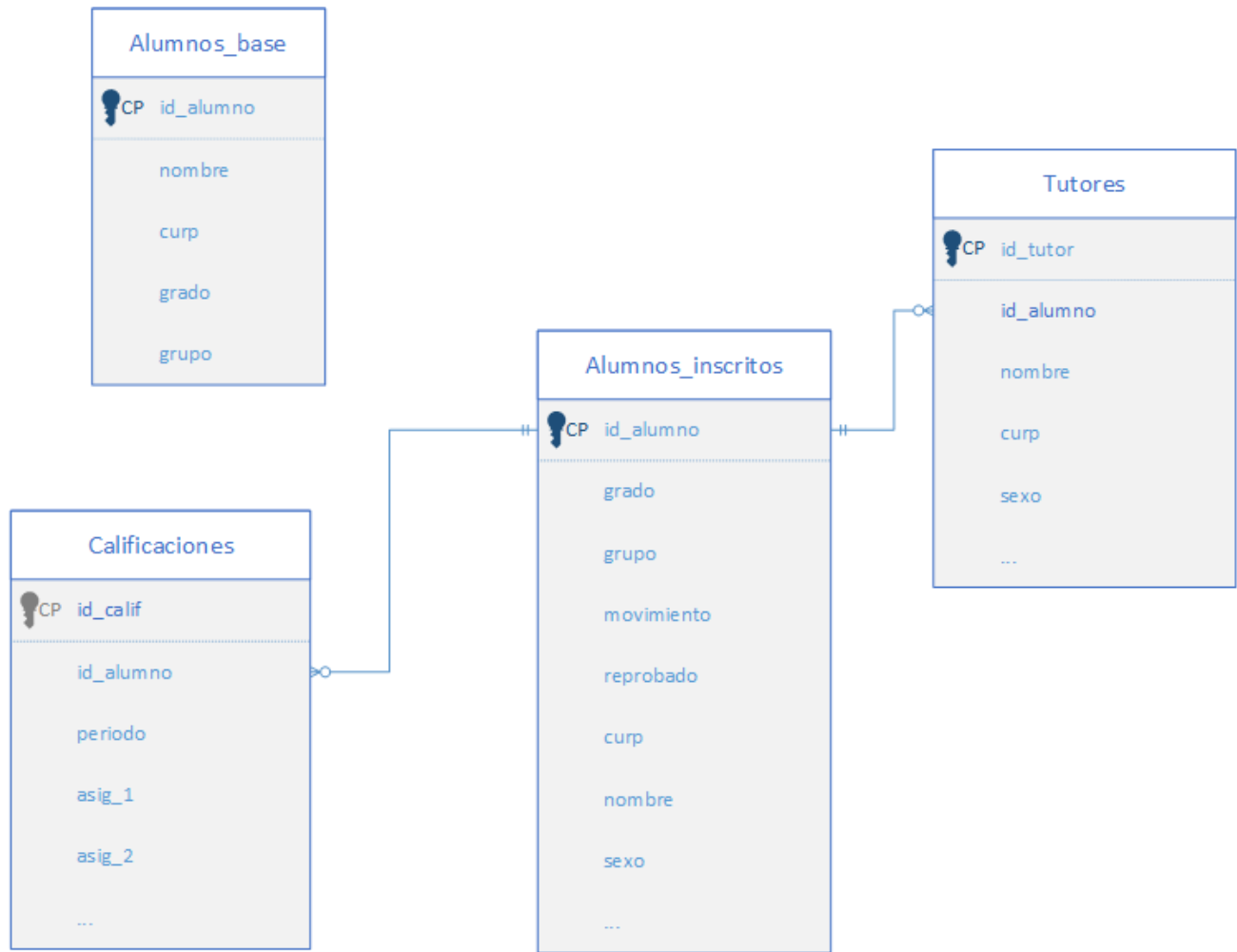


Figura 5.10 Diagrama relacional para el sistema de control escolar



5.4 Diccionario de datos

A continuación, se muestran algunos detalles específicos de las tablas que conformarán la base de datos para el sistema de control escolar, como son el nombre de las columnas, tipo de datos que almacenarán, tamaño de los elementos, una breve descripción de cada atributo, definición de clave primaria o foránea, así como sus valores predeterminados.

Características de la tabla						
Nombre	Alumnos_base					
Descripción	Esta tabla permite el almacenamiento de los datos de los alumnos que han sido previamente autorizados para su inscripción. Permite restringir el acceso solo a los alumnos que estuvieron inscritos el ciclo escolar anterior inmediato y/o alumnos autorizados previamente por la dirección escolar.					
Nombre del atributo	Tipo de dato	Longitud	Descripción	PK	FK	Default
Id_alumno	INT	11	Identificador único	✓		Auto_increment
nombre	VARCHAR	50	Nombre del alumno			
curp	CHAR	18	Clave CURP del alumno			
grado	CHAR	1	Grado escolar			
grupo	CHAR	1	Grupo del alumno			

Cuadro 5.1 Tabla alumnos_base

Características de la tabla						
Nombre	Alumnos_inscritos					
Descripción	Esta tabla permite el almacenamiento de los datos de los alumnos inscritos en el presente ciclo escolar.					
Nombre del atributo	Tipo de dato	Longitud	Descripción	PK	FK	Default



Id_alumno	INT	11	Identificador único	✓	Auto_increment
grado	CHAR	1	Grado escolar		
grupo	CHAR	1	Grupo del alumno		
movimiento	ENUM	"I","A","B"	Movimiento del alumno (inscripción, alta o baja)		I
reprobado	ENUM	"SI","NO"	Promovido o no promovido en el ciclo escolar inmediato anterior.		NO
curp	CHAR	18	Clave CURP		
nombre	VARCHAR	50	Nombre completo del alumno		
sexo	ENUM	"M","H"	Sexo del alumno (Hombre o Mujer)		
Fec_nac	DATE		Fecha de nacimiento		
Entidad_nac	VARCHAR	30	Entidad de nacimiento		
Nacionalidad	VARCHAR	20	Nacionalidad		MEXICANA
Entidad_dom	VARCHAR	30	Entidad de domicilio		MEXICO
Municipio	VARCHAR	30	Nombre del municipio del domicilio		
Localidad	VARCHAR	50	Nombre de la localidad del domicilio		
cp	CHAR	5	Código postal de domicilio		
colonia	VARCHAR	50	Nombre de la colonia o barrio del domicilio		
calle	VARCHAR	34	Nombre de la calle del domicilio		
Entre_calle_1	VARCHAR	35	Nombre de la primera calle entre las que se encuentra el domicilio		
Entre_calle_2	VARCHAR	35	Nombre de la segunda calle entre las que se encuentra el domicilio		
Num_ext	VARCHAR	5	Número exterior de la vivienda		
Num_int	VARCHAR	5	Número interior de la vivienda		



Referencia	VARCHAR	50	Referencia domiciliaria para ubicar mejor el domicilio			
Estatura	VARCHAR	3	Estatura en centímetros			
Peso	VARCHAR	3	Peso en kilogramos			
Tipo_sangre	VARCHAR	3	Tipo de sangre			NR
Zap_orto	ENUM	"NO", "SI"	El alumno utiliza zapato ortopédico			NO
Vacunas_1	ENUM	"NO", "SI"	Vacunas completas hasta los 3 años			NO
Vacunas_2	ENUM	"NO", "SI"	Vacunas completas hasta los 6 años			NO
Vacunas_3	ENUM	"NO", "SI"	Vacunas completas hasta los 11 años			NO
Vacunas_4	ENUM	"NO", "SI"	Vacunas incompletas			NO
Enfermedad_1	ENUM	"NO", "SI"	colesterol			NO
Enfermedad_2	ENUM	"NO", "SI"	cáncer			NO
Enfermedad_3	ENUM	"NO", "SI"	depresión			NO
Enfermedad_4	ENUM	"NO", "SI"	Diabetes			NO
Enfermedad_5	ENUM	"NO", "SI"	Enfermedad renal			NO
Enfermedad_6	ENUM	"NO", "SI"	Lupus			NO
Enfermedad_7	ENUM	"NO", "SI"	Obesidad			NO
Enfermedad_8	ENUM	"NO", "SI"	Problemas cardiovasculares			NO
Enfermedad_9	ENUM	"NO", "SI"	Síndrome metabólico			NO
Enfermedad_10	ENUM	"NO", "SI"	Triglicéridos			NO



apoyo_1	ENUM	“NO”,”SI”	Requiere herramientas de apoyo: dispositivos de ayuda visual			NO
apoyo_2	ENUM	“NO”,”SI”	Requiere herramientas de apoyo: audífonos			NO
apoyo_3	ENUM	“NO”,”SI”	Requiere herramientas de apoyo: ayuda para la movilidad			NO
apoyo_4	ENUM	“NO”,”SI”	Requiere herramientas de apoyo: prótesis			NO
apoyo_5	ENUM	“NO”,”SI”	Requiere herramientas de apoyo: silla de ruedas			NO
alergia	VARCHAR	50	El alumno tiene alguna alergia			
discapacidad	VARCHAR	50	El alumno tiene alguna discapacidad			
Num_tel_1	VARCHAR	10	Número de teléfono principal			
Num_tel_2	VARCHAR	10	Número de teléfono secundario			
Num_tel_emer	VARCHAR	10	Número de teléfono de emergencia			
email	VARCHAR	50	Dirección de correo electrónico			
Red_soc_nombre	VARCHAR	20	Nombre de la red social que usa el alumno			
Red_soc_usuario	VARCHAR	20	Nombre de usuario de la red social que usa el alumno			
Fecha_ins	DATETIME		Fecha y hora de registro			

Cuadro 5.2 Tabla alumnos_inscritos



Características de la tabla						
Nombre	Tutores					
Descripción	Esta tabla permite el almacenamiento de los datos de los tutores de los alumnos en el presente ciclo escolar.					
Nombre del atributo	Tipo de dato	Longitud	Descripción	PK	FK	Default
Id_tutor	INT	11	Identificador único	✓		Auto_increment
Id_alumno	INT	11	Clave foránea que hace referencia a la tabla alumnos_inscritos		✓	
nombre	VARCHAR	50	Nombre completo			
curp	CHAR	18	Clave CURP			
nombre	VARCHAR	50	Nombre completo del tutor			
sexo	ENUM	"M", "H"	Sexo del alumno (Hombre o Mujer)			
Fec_nac	DATE		Fecha de nacimiento			
Entidad_nac	VARCHAR	30	Entidad de nacimiento			
Nacionalidad	VARCHAR	20	Nacionalidad			MEXICANA
Ocupación	VARCHAR	40	Ocupación o profesión			
Estudios	VARCHAR	30	Grado máximo de estudios			
Parentesco	VARCHAR	20	Relación de parentesco con el alumno			
Salario	VARCHAR	30	Ingreso mensual aproximado			
Estado_civil	VARCHAR	15	Estado civil actual del tutor			
email	VARCHAR	50	Dirección de correo electrónico			
Red_soc_nombre	VARCHAR	20	Nombre de la red social que usa el tutor			
Red_soc_usuario	VARCHAR	20	Nombre de usuario de la red social que usa el tutor			
Num_tel	VARCHAR	10	Número de teléfono			



Num_tel_emer	VARCHAR	10	Número de teléfono de emergencia			
--------------	---------	----	----------------------------------	--	--	--

Características de la tabla						
Nombre	Calificaciones					
Descripción	Esta tabla permite el almacenamiento de las calificaciones trimestrales de los alumnos durante el ciclo escolar actual.					
Nombre del atributo	Tipo de dato	Longitud	Descripción	PK	FK	Default
Id_calif	INT	11	Identificador único	✓		Auto_increment
Id_alumno	INT	11	Clave foránea que hace referencia a la tabla alumnos_inscritos		✓	
Periodo	CHAR	1	Periodo o trimestre de calificaciones			
Asig_1	INT	11	Calificación de la asignatura número 1			
Asig_2	INT	11	Calificación de la asignatura número 2			
Asig_3	INT	11	Calificación de la asignatura número 3			
Asig_4	INT	11	Calificación de la asignatura número 4			
Asig_5	INT	11	Calificación de la asignatura número 5			
Asig_6	INT	11	Calificación de la asignatura número 6			
Asig_7	INT	11	Calificación de la asignatura número 7			
Asig_8	INT	11	Calificación de la asignatura número 8			



Asig_8	INT	11	Calificación de la asignatura número 8			
Asig_9	INT	11	Calificación de la asignatura número 9			
Asig_10	INT	11	Calificación de la asignatura número 10			
Inasistencias	INT	11	Inasistencias en el trimestre o periodo			

Cuadro 5.3 Tabla tutores

5.5 Diagramas de flujo

Los diagramas de flujo se utilizan para describir el orden o los pasos que se deben seguir para realizar alguna tarea o procedimiento. Es una representación gráfica de un algoritmo para la resolución de un problema. Su principal ventaja es que representan la información con mucha claridad. También son conocidos como flujogramas u ordinogramas.

Están conformados por una serie de símbolos unidos entre sí por flechas que indican el sentido de la evolución del procedimiento o tarea.

Símbolos principales de un diagrama de flujo.

- Inicio o fin. Un óvalo o elipse representa el inicio o el final de la secuencia de operaciones.

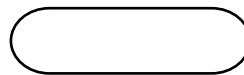


Figura 5.11 Inicio o fin

- Proceso. Un rectángulo representa una operación de proceso, que son la mayor parte de las operaciones realizadas en un programa.



Figura 5.12 Proceso



- Decisión. Un rombo representa una decisión que da lugar a una transferencia condicional de control. Produce una bifurcación entre dos caminos posibles.

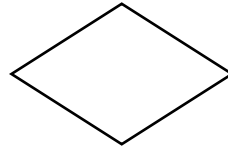


Figura 5.13 Decisión

- Subrutinas. Representa un conjunto de operaciones cuyo empleo se repite varias veces durante la ejecución del programa.



Figura 5.14 Subrutina

- Conectores. Indican el camino que sigue un programa muy largo, es común identificar los círculos mediante un número o letra que sirvan para identificar los conectores que enlazan a una misma trayectoria de flujo.

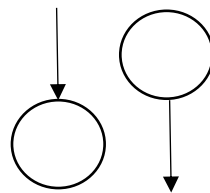


Figura 5.15 Conectores

Las reglas comunes al leer un algoritmo en un diagrama de flujo son:

- Se lee de arriba hacia abajo.
- Las flechas indican el sentido de la lectura o ejecución de cada proceso.
- Solamente hay un inicio y un final en cada diagrama.

5.6 Diagramas de flujo de los procesos del sistema de control escolar

Proceso de captura de datos del usuario tutor.

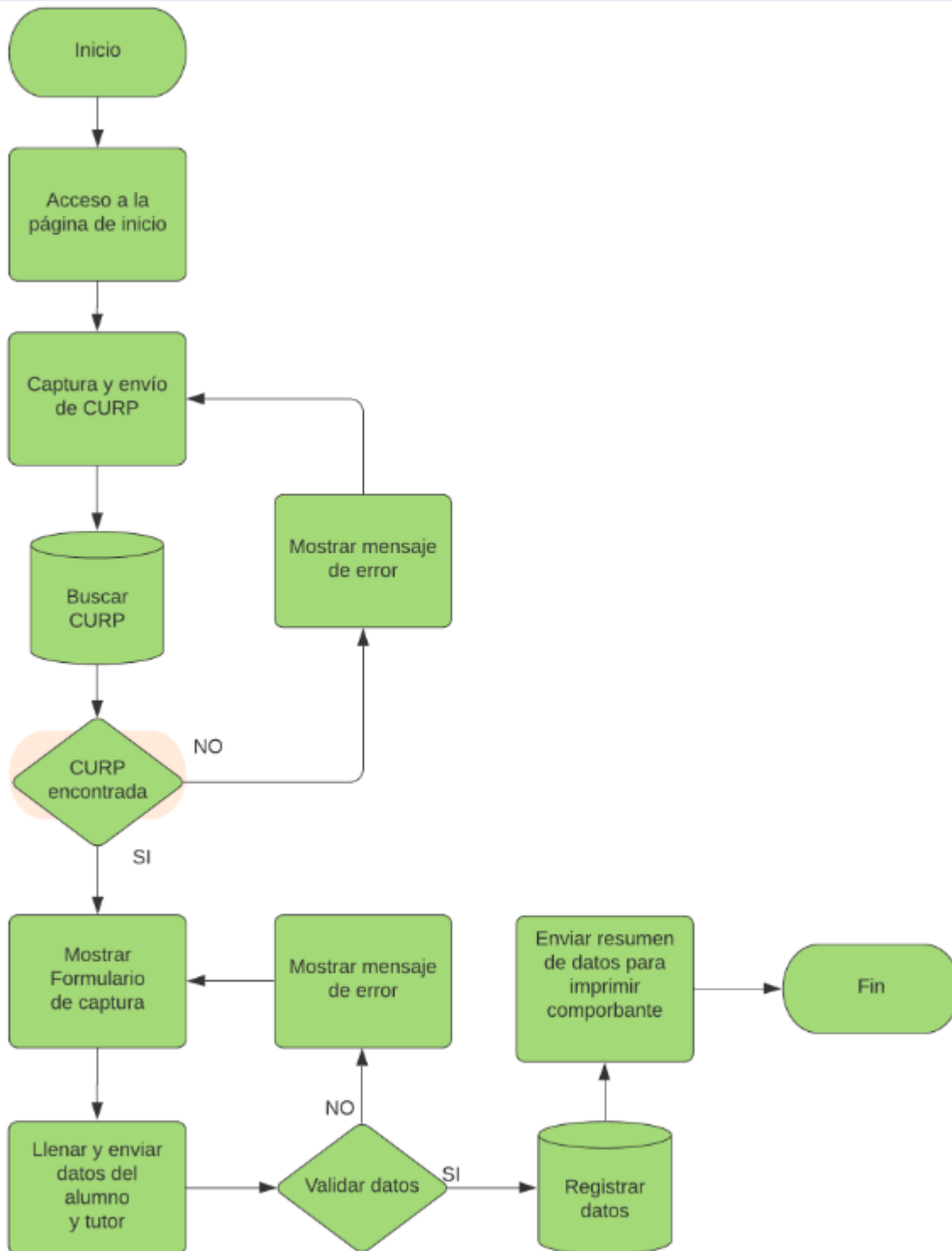


Figura 5.16 Diagrama de flujo del proceso de captura de datos del alumno y tutor



Proceso de captura de calificaciones del usuario docente.

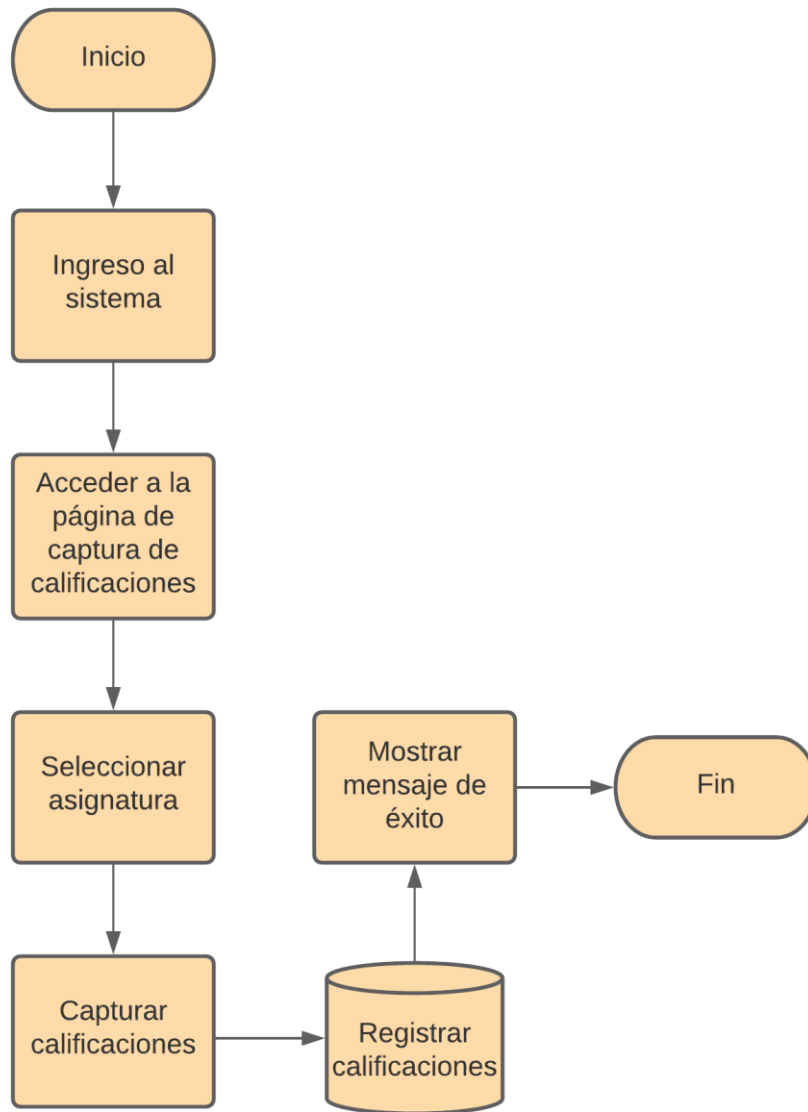


Figura 5.17 Diagrama de flujo del proceso de captura de calificaciones



Proceso de consulta de calificaciones del usuario docente.

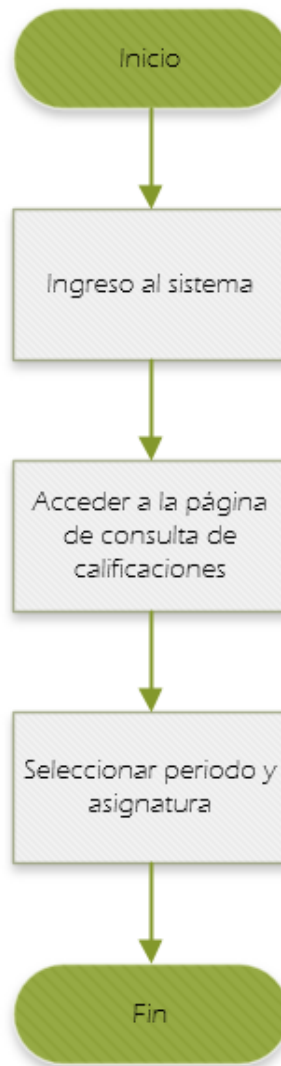


Figura 5.18 Diagrama de flujo del proceso de consulta de calificaciones



Proceso de actualización de calificaciones del usuario docente.

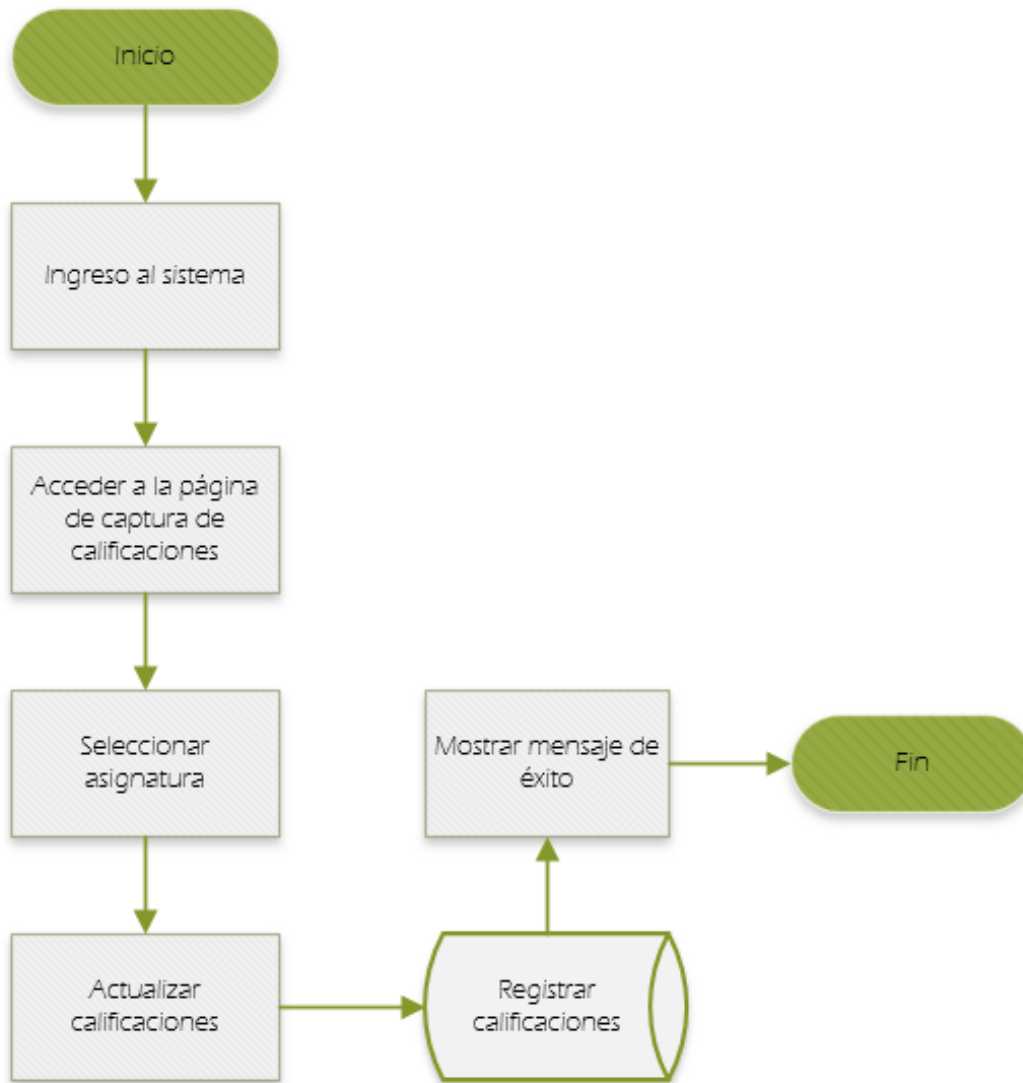


Figura 5.19 Diagrama de flujo del proceso de actualización de calificaciones



Proceso de consulta datos del alumno del usuario docente.

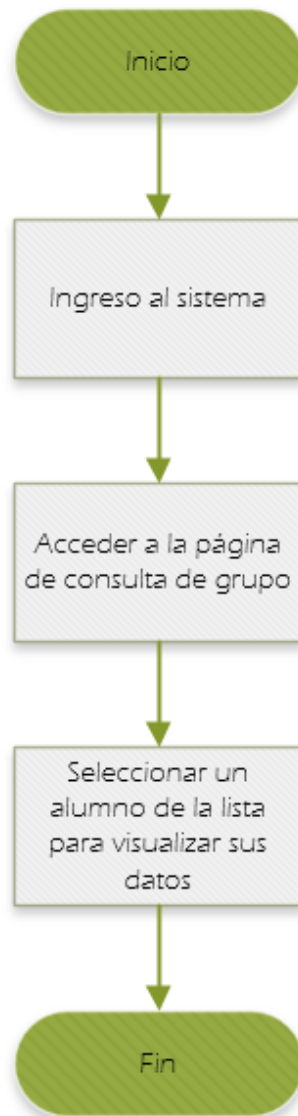


Figura 5.20 Diagrama de flujo del proceso de consulta de datos del alumno



Proceso de consulta datos del tutor del usuario docente.

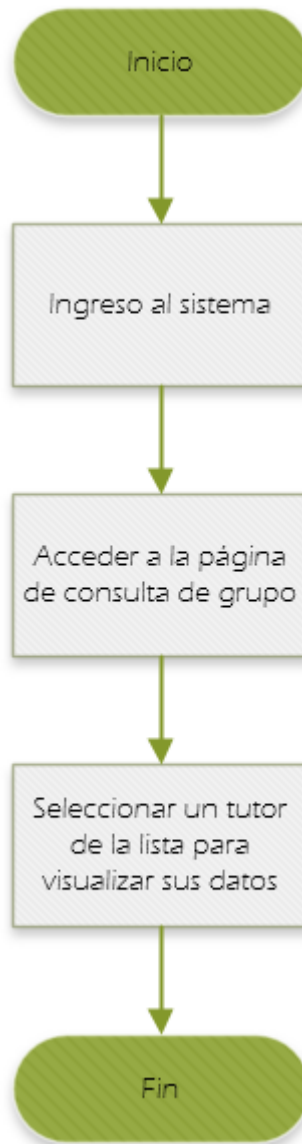


Figura 5.21 Diagrama de flujo del proceso de consulta de datos del tutor



Proceso de consulta de datos de alumnos del usuario administrador.

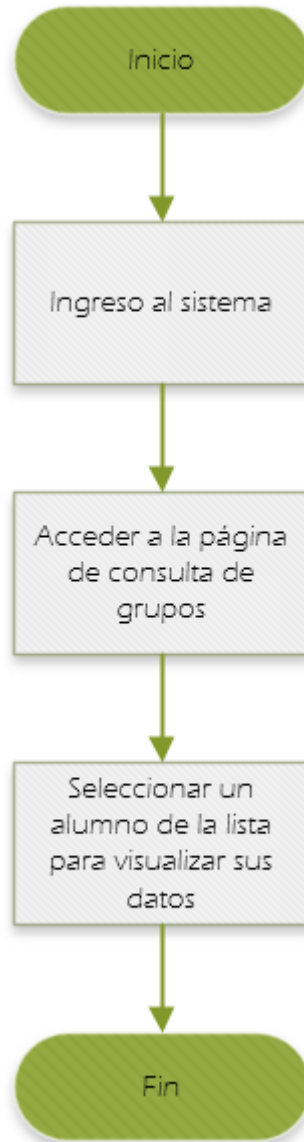


Figura 5.22 Diagrama de flujo del proceso de consulta de datos del alumno



Proceso de consulta de datos del tutor del usuario administrador.

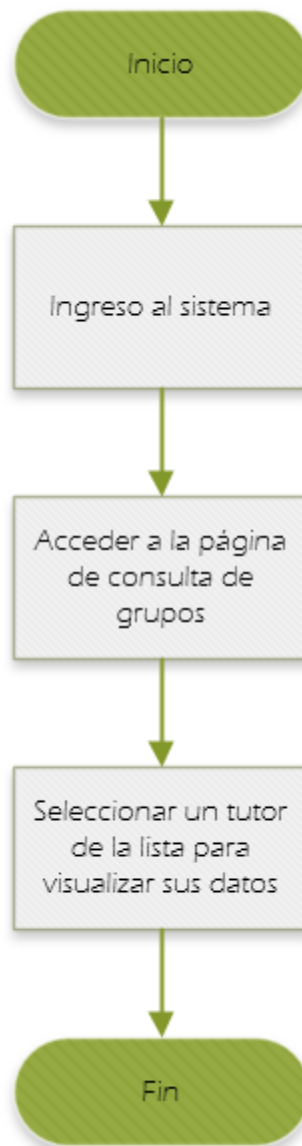


Figura 5.23 Diagrama de flujo del proceso de consulta de datos del tutor

Proceso de actualización de datos de un alumno o tutor del usuario administrador.

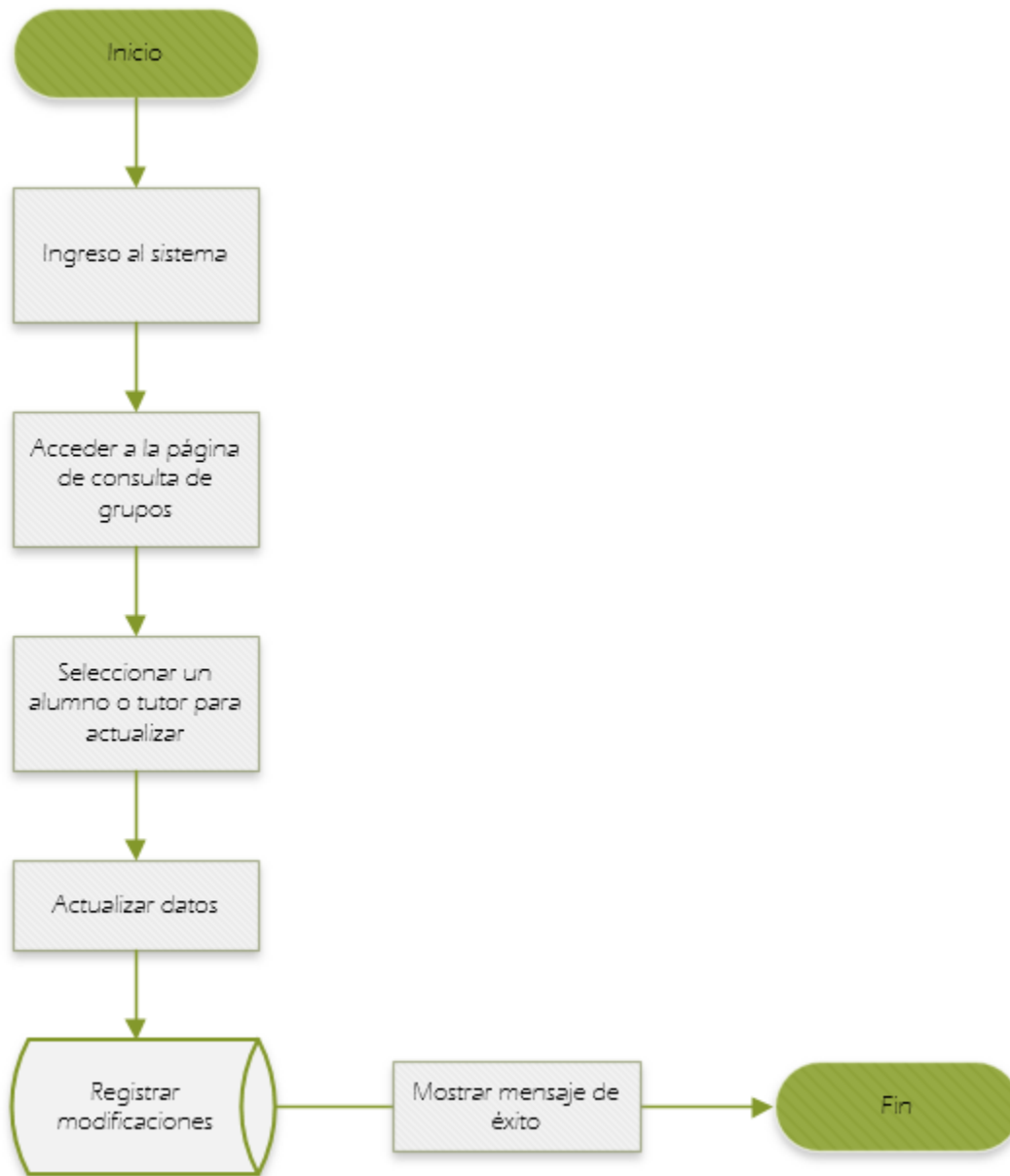


Figura 5.24 Diagrama de flujo del proceso de actualización de datos de alumnos o tutores



Proceso de consulta de calificaciones por grupo del usuario administrador.

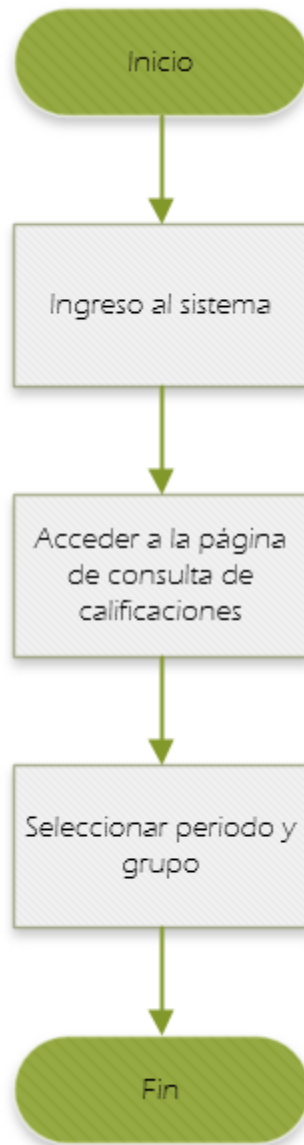


Figura 5.25 Diagrama de flujo del proceso de consulta calificaciones por grupo



Proceso de ingreso de nuevos alumnos en el sistema del usuario administrador.

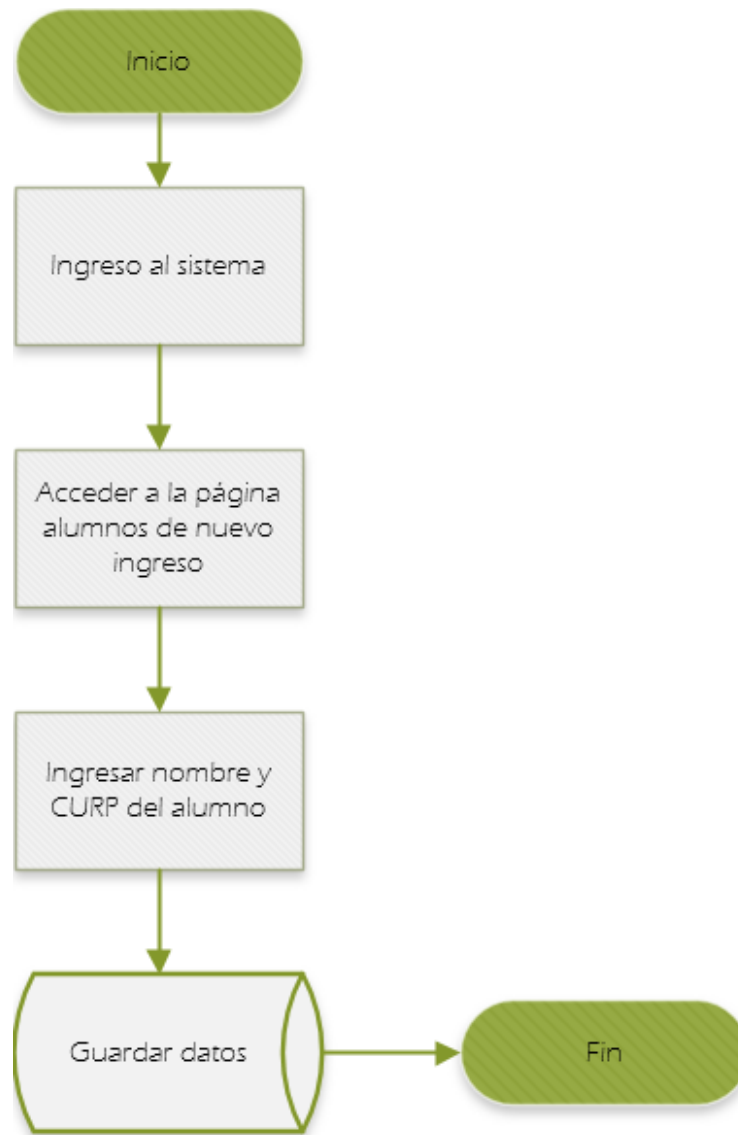


Figura 5.26 Diagrama de flujo del proceso de alta de nuevo alumno

Proceso de eliminación de un alumno del usuario administrador.

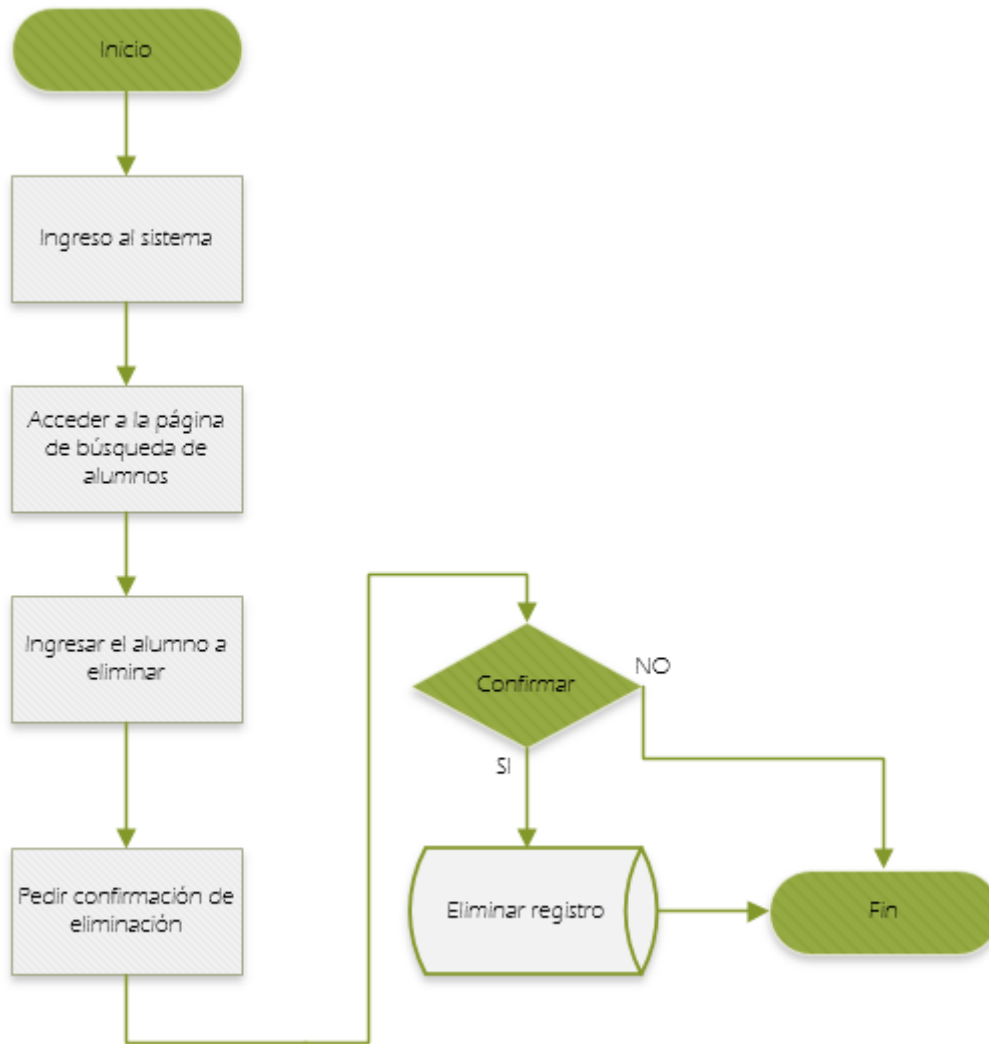


Figura 5.27 Diagrama de flujo del proceso de eliminación de un alumno



CAPÍTULO 6. DESARROLLO



6.1 Servidor web

Un servidor web es un programa que sirve para atender y responder a las distintas peticiones que hacen los navegadores proporcionando los recursos que estos soliciten a través de los protocolos HTTP o HTTPS. Un servidor web básico cuenta con un esquema de funcionamiento muy simple, que se basa en ejecutar indefinidamente el siguiente proceso:

- Esperar peticiones en un puerto TCP (Transfer Control Protocol).
- Recibir una petición.
- Buscar un recurso.
- Enviar el recurso usando la misma conexión por la que recibió la petición.
- Volver al punto inicial

6.2 PHP

PHP es acrónimo recursivo para “PHP: Hypertext Preprocessor” es un lenguaje de programación interpretado de código abierto y muy popular en el desarrollo web. Tiene una sintaxis muy similar a C/C++ y Perl.

PHP permite incluir sus instrucciones intercaladas dentro del código HTML, para generar documentos dinámicos. Dentro del servidor web las instrucciones PHP se transforman en código de texto HTML y son enviadas al cliente como un documento HTML ordinario.

6.3 HTML

Es un lenguaje de etiquetas estandarizado por la W3C (World Wide Web Consortium) utilizado para desarrollar páginas y documentos web. Dichas etiquetas especiales son intercaladas en un documento de texto plano que serán interpretadas posteriormente por un navegador web. La versión más reciente de este lenguaje es HTML5 que agrega mucho más dinamismo y mejoras al lenguaje.



Estructura básica de una página en HTML5.

```
1 <!DOCTYPE html>
2 <html lang="es">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Título del documento</title>
7 </head>
8
9 <body>
10
11 </body>
12 </html>
13
```

Principales etiquetas para la estructura de un documento HTML5	
Etiqueta HTML	Descripción
<!DOCTYPE>	Se utiliza para especificar el tipo de documento.
<html>	Define y establece el inicio en un documento HTML.
<head>	Indica el comienzo del encabezado del documento.
<title>	Permite indicar el título del documento. Se ubica en el encabezado del documento.
<meta>	Con esta etiqueta, es posible incluir información de metadatos en el documento. Por ejemplo, descripción, palabras clave, autor, etcétera.
<link>	Se utiliza para vincular archivos externos. Define una relación entre el documento actual y una fuente externa (por ejemplo, un archivo CSS). Se ubica en el encabezado del documento
<style>	Se utiliza para definir los estilos CSS del documento.
<script>	Contiene los scripts del lado cliente del documento. Si se utiliza el atributo src , es posible indicar la ruta de un archivo de script externo.
<body>	Marca el comienzo del cuerpo del documento.

Cuadro 6.1 Etiquetas HTML básicas



6.4 MYSQL

Es un sistema gestor de base de datos relacionales (SGBD) multiplataforma y de código abierto. Es un sistema cliente – servidor que consiste en un servidor SQL multihilo que trabaja con una variedad de diferentes programas y bibliotecas cliente, herramientas administrativas y una gran cantidad de interfaces de programación para aplicaciones (API).

6.5 MariaDB

MariaDB es un sistema gestor de base de datos relacionales (SGBD) multiplataforma y de código abierto que surge como bifurcación de MySQL. Todos los comandos, interfaces, librerías, APIs de clientes y nombres de archivos son exactamente iguales que en MySQL.

6.6 Servidor de pruebas locales con XAMPP

XAMPP es un entorno de desarrollo de software libre que contiene el servidor apache, el gestor de base de datos MariaDB, así como el lenguaje de programación PHP. Este entorno de desarrollo es útil para realizar todas las pruebas de la aplicación de manera local antes de desplegarla en un servicio de hosting web profesional.

Trabajar de manera local en el desarrollo de la aplicación resulta ser muy práctico, ya que se puede probar el código localmente, sin necesidad de transferir los archivos al servicio de web hosting, lo que conlleva a un ahorro de tiempo considerable.

Una vez instalado el entorno XAMPP, es posible probar cada una de las páginas codificadas en el lenguaje php o html, simplemente anteponiendo en la barra de direcciones del navegador la dirección ip del propio equipo. Alternativamente, también es posible acceder al servidor de pruebas escribiendo la palabra “localhost” antes del directorio de la aplicación.

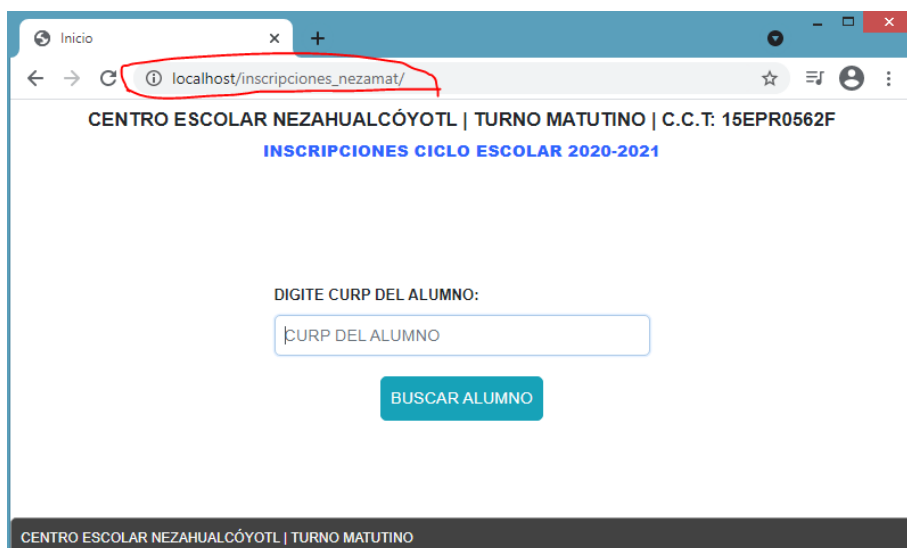
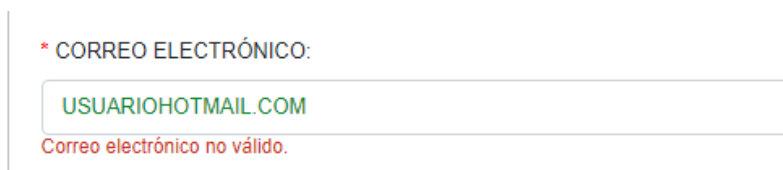


Figura 6.1 Pruebas en el servidor local XAMPP

6.7 Validación de los campos de formularios.

Validetta JS. Es una pequeña librería JavaScript que sirve para llevar a cabo la validación de campos de formulario del lado del cliente. Es de código abierto, ocupa muy poco espacio y permite comprobar campos vacíos, de correo electrónico, numéricos, campos iguales entre sí, etcétera. En caso de existir algún error al momento de rellenar los campos, Validetta JS enviará un mensaje de ayuda al usuario.



En el siguiente ejemplo se hace obligatorio el llenado del campo, ya que contiene la instrucción data-validetta con el atributo = "required":

```
<input type= "text" id="nombre" name="nombre" data-validetta="required" />
```

En este otro ejemplo, también es obligatorio llenar el campo, pero además se comprueba que el formato para la dirección de correo electrónico sea el correcto:



```
<input type="text" id="email" name="email" data-validation="required,email" />
```

Por otro lado, cuando los datos del formulario son enviados al servidor, también se lleva a cabo una validación en cuanto llegan los datos provenientes del lado del cliente, en el siguiente ejemplo, se comprueba que la variable recién llegada se encuentre definida y que no esté vacía:

```
if(isset($_POST['nombre']) && $_POST['nombre'] <> "") {  
  
}
```

También se comprueban en casos específicos los tipos de datos esperados, por ejemplo, aquí se comprueba que la variable contenga un tipo de dato numérico:

```
if(is_numeric($_POST['calificacion'])) {  
  
}
```

6.9 Creación de las tablas que conforman la base de datos.

A continuación, se muestra el código para la creación de las tablas `alumnos_base`, `alumnos_inscritos`, `tutores` y `calificaciones`:

Tabla `alumnos_base`.

```
1 CREATE TABLE alumnos_base (  
2 id_alumno INT AUTO_INCREMENT NOT NULL PRIMARY KEY,  
3 nombre VARCHAR(50) NOT NULL,  
4 curp VARCHAR(20) NOT NULL,  
5 grado CHAR(1),  
6 grupo CHAR(1),  
7 INDEX indice_curp (curp))  
8 engine=innodb DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci
```



Tabla `alumnos_inscritos`.

```
CREATE TABLE alumnos_inscritos (  
  id_alumno INT AUTO_INCREMENT NOT NULL PRIMARY KEY,  
  grado CHAR(1) NOT NULL,  
  grupo CHAR(1) NOT NULL,  
  movimiento ENUM('I', 'A', 'B') NOT NULL DEFAULT 'I',  
  reprobado ENUM('NO', 'SI') NOT NULL DEFAULT 'NO',  
  curp CHAR(18) NOT NULL,  
  nombre VARCHAR(50) NOT NULL,  
  sexo ENUM('M', 'H') NOT NULL,  
  fec_nac DATE NOT NULL,  
  entidad_nac VARCHAR(30) NOT NULL,  
  nacionalidad VARCHAR(20) NOT NULL DEFAULT 'MEXICANA',  
  entidad_dom VARCHAR(30) NOT NULL DEFAULT 'MEXICO',  
  municipio VARCHAR(30) NOT NULL,  
  localidad VARCHAR(50) NOT NULL,  
  cp CHAR(5) NOT NULL,  
  colonia VARCHAR(50) NOT NULL,  
  calle VARCHAR(35) NOT NULL,  
  entre_calle_1 VARCHAR(35),  
  entre_calle_2 VARCHAR(35),  
  num_ext VARCHAR(5) NOT NULL,  
  num_int VARCHAR(5),  
  referencia VARCHAR(50) NOT NULL,  
  estatura VARCHAR(3),  
  peso VARCHAR(3),  
  tipo_sangre VARCHAR(3) DEFAULT 'NR',  
  zap_orto ENUM('NO', 'SI') NOT NULL DEFAULT 'NO',  
  lentes ENUM('NO', 'SI') NOT NULL DEFAULT 'NO',
```




```
vacunas_1 ENUM('NO', 'SI') NOT NULL DEFAULT 'NO',
vacunas_2 ENUM('NO', 'SI') NOT NULL DEFAULT 'NO',
vacunas_3 ENUM('NO', 'SI') NOT NULL DEFAULT 'NO',
vacunas_4 ENUM('NO', 'SI') NOT NULL DEFAULT 'NO',
enfermedad_1 ENUM('NO', 'SI') NOT NULL DEFAULT 'NO',
enfermedad_2 ENUM('NO', 'SI') NOT NULL DEFAULT 'NO',
enfermedad_3 ENUM('NO', 'SI') NOT NULL DEFAULT 'NO',
enfermedad_4 ENUM('NO', 'SI') NOT NULL DEFAULT 'NO',
enfermedad_5 ENUM('NO', 'SI') NOT NULL DEFAULT 'NO',
enfermedad_6 ENUM('NO', 'SI') NOT NULL DEFAULT 'NO',
enfermedad_7 ENUM('NO', 'SI') NOT NULL DEFAULT 'NO',
enfermedad_8 ENUM('NO', 'SI') NOT NULL DEFAULT 'NO',
enfermedad_9 ENUM('NO', 'SI') NOT NULL DEFAULT 'NO',
enfermedad_10 ENUM('NO', 'SI') NOT NULL DEFAULT 'NO',
apoyo_1 ENUM('NO', 'SI') NOT NULL DEFAULT 'NO',
apoyo_2 ENUM('NO', 'SI') NOT NULL DEFAULT 'NO',
apoyo_3 ENUM('NO', 'SI') NOT NULL DEFAULT 'NO',
apoyo_4 ENUM('NO', 'SI') NOT NULL DEFAULT 'NO',
apoyo_5 ENUM('NO', 'SI') NOT NULL DEFAULT 'NO',
otro_apoyo VARCHAR(30),
alergia VARCHAR(50),
discapacidad VARCHAR(50),
num_tel_1 VARCHAR(10) NOT NULL,
num_tel_2 VARCHAR(10),
num_tel_emer VARCHAR(10),
email VARCHAR(50) NOT NULL,
red_soc_nombre VARCHAR(20),
red_soc_usuario VARCHAR(20),
fecha_ins DATETIME,
INDEX indice_nombre (nombre))
engine=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci
```



Tabla tutores.

```
CREATE TABLE tutores (  
  id_tutor INT AUTO_INCREMENT NOT NULL PRIMARY KEY,  
  id_alumno INT NOT NULL,  
  nombre VARCHAR(50) NOT NULL,  
  curp CHAR(18) NOT NULL,  
  sexo ENUM('M', 'H') NOT NULL,  
  fec_nac DATE NOT NULL,  
  entidad_nac VARCHAR(30),  
  nacionalidad VARCHAR(20) DEFAULT 'MEXICANA',  
  ocupacion VARCHAR(40) NOT NULL,  
  estudios VARCHAR(30) NOT NULL,  
  parentesco VARCHAR(20) NOT NULL,  
  salario VARCHAR(30) NOT NULL,  
  estado_civil VARCHAR(15) NOT NULL,  
  email VARCHAR(50) NOT NULL,  
  red_soc_nombre VARCHAR(20),  
  red_soc_usuario VARCHAR(30),  
  num_tel VARCHAR(10),  
  num_tel_emer VARCHAR(10),  
  INDEX indice_nombre (nombre),  
  FOREIGN KEY (id_alumno) REFERENCES alumnos_inscritos (id_alumno) ON DELETE CASCADE)  
engine=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci
```



Tabla calificaciones.

```
CREATE TABLE calificaciones (  
    id_calif INT AUTO_INCREMENT NOT NULL PRIMARY KEY,  
    id_alumno INT NOT NULL,  
    periodo CHAR(1) NOT NULL,  
    asig_1 INT NULL,  
    asig_2 INT NULL,  
    asig_3 INT NULL,  
    asig_4 INT NULL,  
    asig_5 INT NULL,  
    asig_6 INT NULL,  
    asig_7 INT NULL,  
    asig_8 INT NULL,  
    asig_9 INT NULL,  
    asig_10 INT NULL,  
    inasistencias INT NULL DEFAULT 0,  
    FOREIGN KEY (id_alumno) REFERENCES alumnos_inscritos (id_alumno) ON DELETE CASCADE)  
engine=innnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci
```

6.10 Conexión a MySQL desde PHP

Para la conexión a la base de datos MySQL, se utilizó una extensión de PHP llamada MySQLi, primero se definieron los datos de conexión (host, usuario, password y el nombre de la base de datos) en un archivo PHP por separado llamado config.php.

```
1 <?php  
2  
3 define('DB_HOST', 'localhost');  
4 define('DB_USER', 'root');  
5 define('DB_PASS', '');  
6 define('DB_NAME', 'inscripciones');  
7 define('DB_CHARSET', 'utf-8');  
8  
9 ?>
```



Después, en otro archivo, se creó una clase llamada `CconMysql` que requiere que se incluyan los datos de conexión del archivo anterior, esto se logra con la instrucción `require_once` "config.php":

```
1 <?php
2 require_once "config.php";
3 class CconMysql
4 {
5     protected $database;
6     public function __construct() {
7         $this->database = new mysqli(DB_HOST, DB_USER, DB_PASS, DB_NAME);
8         if ( $this->database->connect_errno ) {
9             echo "Error al conectar a MySQL: ". $this->database->connect_error;
10        }
11        else {
12            $this->database->query("SET NAMES 'utf8'");
13        }
14    }
15 }
16 ?>
17
```

En el constructor de la clase `CconMysql` se crea una instancia de la clase `mysqli` que representa una conexión a la base de datos y que es guardada en una propiedad llamada `$database` que será devuelta al instanciar la clase `CconMysql` en cualquier lugar. Esta clase se utilizará para conectar a la base de datos MySQL y realizar las consultas en todas las páginas de la aplicación.

Ejemplo de definición de un método para la consulta de datos en donde se crea una nueva clase llamada `CGestionData` que hereda de la clase `CconMysql` y que invoca al constructor de la clase padre (`CconMysql`) para obtener una conexión a la base de datos.



```
1 <?php
2 require_once "CconMysql.php";
3 class CGestionData extends CconMysql {
4
5     public function __construct()
6     {
7         parent::__construct(); // Invoca al constructor de la clase padre (CconMysql)
8     }
9
10    public function existeAlumnoBase($curp)
11    {
12        $resultado = $this->database->query("SELECT count(*) FROM alumnos_base WHERE curp='$curp'");
13        return $resultado;
14    }
15 }
16
17 ?>
```

6.11 Sesiones para acceso a los usuarios.

Para proporcionar el acceso seguro a la aplicación, se hace uso de variables de sesión en todas las páginas de la aplicación, para garantizar que solo pueda ingresar el personal autorizado que cuente con el nombre de usuario y la contraseña.

Primero se establecen el nombre de usuario y la contraseña arbitrariamente y de acuerdo con las indicaciones de la dirección escolar, por ejemplo, para el ingreso del usuario administrador, primero se muestra un formulario en donde se pide ingresar el nombre de usuario y una contraseña:

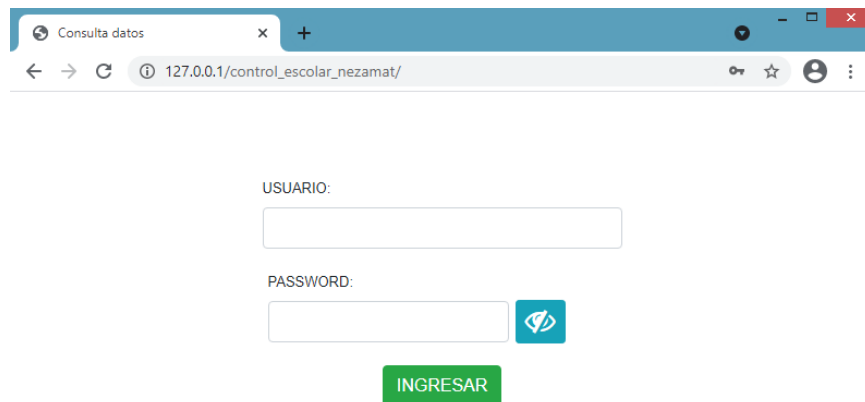


Figura 6.2 Acceso para el administrador



El código HTML del formulario (de forma sintetizada) es el siguiente:

```
<form action="index.php" method="post">
<input type="text" id="user" name="user">
<input type="password" id="pwd" name="pwd">
<button type="submit" name="comprobar" >INGRESAR</button>
</form>
```

Enseguida de introducir las credenciales, se hace clic en el botón submit del formulario, como el atributo action de la etiqueta form, apunta a la misma página (index.php) se provoca una recarga, entonces se ejecuta el siguiente código PHP:

```
<?php
if(isset($_POST['comprobar']) && $_POST['user'] == "15epr0562f") {
    $pwd = $_POST['pwd'];
    If($pwd == 'zY6i8SMv') {
        $_SESSION["user"] = "admin";
        header ("Location: buscar_alumno.php");
    }
?>
```

Cuando la variable `$_POST['comprobar']` está definida (lo cual ocurre cuando es presionado el botón de envío submit) y la variable `$_POST['user']` tiene como valor "15epr0562f" (usuario proporcionado por la dirección escolar), se comprueba con una estructura if, que el



valor de la variable \$pwd sea 'zY6i8SMv', si este es el caso, entonces se crea una variable de sesión para identificar al usuario como "admin" en todas las páginas de la aplicación.

```
$_SESSION["user"] = "admin";
```

Enseguida, con la función header() de PHP, se envía al usuario a la página de inicio de la aplicación llamada buscar_alumno.php.

```
header ("Location: buscar_alumno.php");
```

Para comprobar que el usuario sea el correcto, en cada una de las páginas que conforman la aplicación se agrega el siguiente código:

```
<?php  
session_start();  
if($_SESSION["user"] == "admin")  
{  
?>  
  
// Código principal de la página  
  
<?php  
}  
else {  
    echo "<h1>Zona restringida</h1>";  
    echo "<h2><a href='index.php'>Iniciar session</a></h2>";  
}  
?>
```



De esta manera, primero se comprueba que la variable de sesión (user) sea “admin” para poder acceder al contenido de la página, en caso contrario, solo se mostrará un aviso de que se encuentran en una zona restringida y se les invitará a iniciar sesión con un enlace que los dirige a la página para hacer el log in.

6.12 Protocolo IP

El protocolo IP es el protocolo más básico de Internet y proporciona todos los servicios necesarios para el transporte de datos como los siguientes:

- **Direccionamiento.** Las cabeceras IP contienen las direcciones de las máquinas de origen y destino, mismas que serán usadas por los enrutadores para decidir el tramo de red por el que circularán.
- **Fragmentación.** Si la información a transmitir es demasiado grande, puede ser dividida en paquetes más pequeños para posteriormente ser reensamblada.
- **Tiempo de vida del paquete.** Cada paquete IP contiene un valor de tiempo de vida, que disminuye cada que el enrutador recibe y reenvía el paquete. Cuando este valor es cero, el paquete se pierde y deja de ser reenviado.
- **Tipo de servicio.** Puede indicar la prioridad del paquete.

6.13 Dirección IP

Una dirección IP es un número que identifica de manera lógica y jerárquica a una interfaz de un dispositivo, por ejemplo, una tablet, un celular o una computadora, dentro de una red que utilice el protocolo de Internet (Internet Protocol). Las direcciones IP se pueden expresar con números de notación decimal dividiendo los 32 bits de la dirección en cuatro octetos. El valor decimal de cada octeto puede ser entre 0 y 255. Por ejemplo, esta es una dirección IPv4 de un dispositivo conectado a una red local: 192.168.0.9



La principal diferencia entre IPv4 e IPv6 es que IPv6 admite direcciones de internet de 128 bits (mucho más grande), mientras que IPv4 lo hace de 32 bits.

Los sitios de Internet que requieren estar conectados permanentemente cuentan con una dirección IP fija o estática, que no cambia con el tiempo, ya que esto facilita su ubicación en la red. Los humanos, sin embargo, utilizamos una notación mucho más fácil de recordar y utilizar como los nombres de dominio (DNS) por sus siglas en inglés.

6.14 Servicio de identificación de dominios

El servicio de nombres de dominio (DNS) por sus siglas en inglés, es una forma sencilla de localizar un dispositivo en Internet, se identifica por su dirección IP. Debido a que las personas recuerdan más fácilmente nombres que números, se desarrolló un sistema que puede convertir los números de la dirección IP en un nombre fácil de recordar. Los nombres de dominio sirven además para proporcionar una identificación propia en la red.

Un servidor de nombre de dominios busca una dirección IP a partir del nombre de una computadora, y también puede encontrar su nombre a partir de su dirección IP.

6.15 Servicio de web hosting

Un servicio de alojamiento o web hosting permite desplegar nuestra aplicación o sitio web en una o varias computadoras conectadas las 24 horas del día los 365 días del año, en el caso del Centro Escolar, después de haber investigado y cotizado costos de algunos proveedores del servicio, se decidió por contratar el servicio en una empresa llamada Xpress Hosting, que ofrece básicamente las siguientes características:

- Soporte para bases de datos MySQL (16 gigabytes de capacidad de almacenamiento).
- Soporte para PHP distintas versiones.
- Un dominio sin costo durante el primer año (La empresa proporciona la configuración del dominio gratuitamente).
- Costo de \$770.00 MXN un año.



El nombre del dominio (DNS) para la aplicación de control escolar es:

<http://cescolarnezatm.com>

6.16 Administración del sitio

Al momento de contratar el servicio de alojamiento, la empresa proporciona las credenciales necesarias para establecer conexión vía FTP (File Transfer Protocol) con el servidor y administrar los archivos necesarios. Alternativamente es posible administrar el sitio a través de una herramienta web llamada plesk.

FTP

FTP (File Transfer Protocol) o protocolo de transferencia de archivos es un protocolo de red para la transferencia de archivos entre sistemas conectados a una red TCP (Transmission Control Protocol), basado en la arquitectura cliente -servidor. Desde un equipo cliente es posible conectar a un servidor para descargar o enviar archivos independientemente del sistema operativo utilizado en cada computadora.

El uso más común de un servidor FTP es el alojamiento web, en el que sus clientes utilizan el servicio para subir archivos de páginas web. El servicio FTP es ofrecido por la capa de aplicación del modelo TCP/IP al usuario utilizando normalmente el puerto 20 o 21.

Cliente FTP

Un cliente FTP permite la conexión a un servidor FTP para realizar intercambio de archivos. Existen distintos tipos de clientes FTP, desde los que utilizan una terminal del sistema operativo en modo texto, hasta los que utilizan una interfaz gráfica como FileZilla.

FileZilla

FileZilla es un cliente FTP de código abierto y multiplataforma, además es muy fácil de utilizar. Una vez establecida la conexión con el servidor FTP, la aplicación muestra



básicamente dos columnas, en la columna de la izquierda que representa el equipo local, se muestra un navegador de archivos desde el cual se pueden seleccionar elementos para subir al servidor remoto que se visualiza en la columna de la derecha. También es posible descargar elementos desde el servidor remoto hacia el equipo local.

Transferencia de archivos al servidor remoto utilizando el cliente FTP FileZilla.

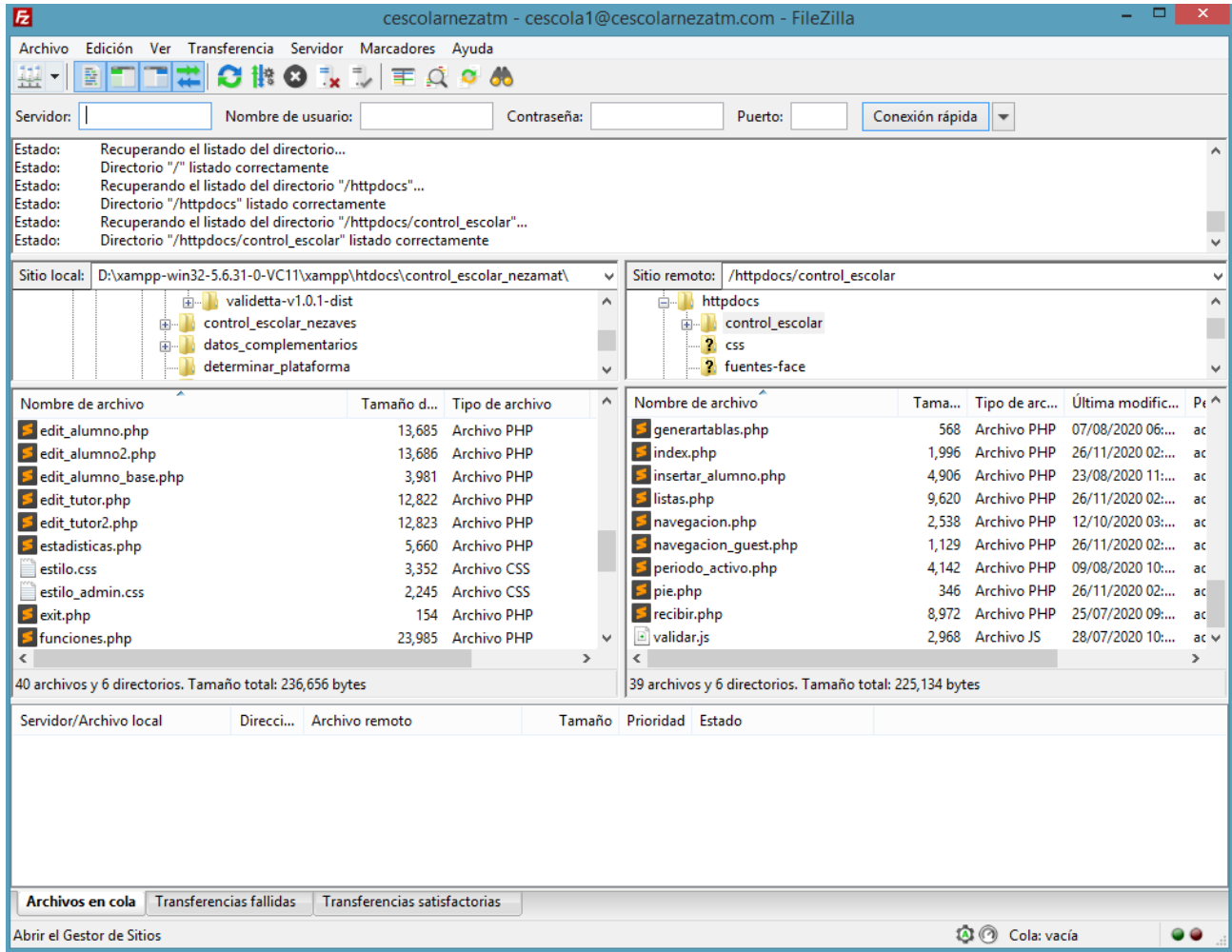


Figura 6.3 Cliente FTP FileZilla



Conclusiones

Gracias a la disposición y colaboración del personal involucrado en el proyecto para el desarrollo del Sitio Web “COES-NEZA” (Control Escolar Nezahualcóyotl), se lograron captar en esencia los requerimientos del sistema y de los usuarios, dando como resultado, que la aplicación web desarrollada cumpla con el objetivo principal, automatizando la información para reducir al máximo el tiempo necesario para completar el proceso de inscripciones, además de hacerlo de una forma eficiente, confiable y segura.

El COES-NEZA ha sido de gran ayuda en el proceso de recopilación de datos de alumnos y tutores, gracias a que se encuentra alojado en un servidor web profesional que está en línea las 24 horas del día, los usuarios pueden acceder en cualquier momento y desde cualquier lugar, utilizando una computadora, tablet o smartphone y una conexión a Internet. Esto cobra mayor importancia si se considera que el proceso de obtención de datos a través del COES-NEZA, evita el desplazamiento físico de los padres de familia al centro escolar, ahorrando tiempo y esfuerzo en dicho proceso, evitando además la concentración física de personas en las aulas de la escuela en donde se acostumbraba realizar las inscripciones presenciales.

El proceso de inscripción a través del COES-NEZA ha permitido inscribir a los alumnos de manera remota, evitando la conglomeración y el riesgo de contagio por COVID-19.

Por otro lado, desde el punto de vista de la dirección escolar, el COES-NEZA permite que el personal encargado del manejo de los datos, pueda tener acceso a la información un instante después de que ésta ha sido ingresada. Pudiendo monitorear en tiempo real dicha actividad y observar quiénes han realizado su registro y quienes aún no.

Por su parte, el personal de la dirección escolar puede consultar información relevante de una manera sencilla, organizada y al instante, reduciendo los tiempos de espera en la búsqueda de información, lo que deriva en una administración escolar mucho más eficiente.

En cuanto a la concentración de las calificaciones de los alumnos, es importante destacar que anteriormente no existía una única forma de hacerlo, algunos docentes enviaban las calificaciones a la dirección escolar utilizando distintos métodos y formatos, por ejemplo, podían reportar las calificaciones trimestrales a través de correo electrónico, o utilizando



mensajería instantánea como WhatsApp, usando un formato en procesador de texto, en una hoja de cálculo, o incluso, en una fotografía de un formato llenado a mano, es decir, no existía una manera unificada de hacerlo, sino que cada docente utilizaba el método que le parecía más sencillo o con el que más estaba familiarizado. El COES-NEZA resuelve esta problemática concentrando las calificaciones organizadas por asignatura y grupo en un solo lugar, facilitando la captura de calificaciones y consulta de datos de los alumnos y tutores a su cargo, desde cualquier dispositivo conectado a Internet en el momento que ellos lo necesiten. A su vez la dirección escolar percibe una importante disminución de carga administrativa, al disponer de las calificaciones trimestrales concentradas automáticamente en un solo lugar y listas para ser consultadas.

Finalmente, la primera versión del COES-NEZA, ha demostrado su utilidad al inicio de este ciclo escolar 2020-2021, proporcionando los beneficios inherentes de una aplicación web y resolviendo las problemáticas principales que señalaron sus usuarios en el inicio de este proyecto. Se espera en un futuro continuar con la actualización de esta aplicación, agregando mejoras y ampliando funcionalidades para beneficio de toda la comunidad escolar.



Anexos

Manuales de usuario

Los manuales de usuario proporcionan una guía para que el usuario pueda utilizar de manera correcta la aplicación, en ellos se explica a detalle, paso a paso y con ayuda visual los pasos que se deben seguir para realizar alguna tarea dentro de la aplicación web de control escolar. Existen tres tipos de usuario según los casos de uso a saber:

- Tutor o padre de familia
- Docente
- Administrador

Cada usuario tiene distintos roles y debe realizar diferentes actividades dentro de la aplicación, es por eso que se elaboró un manual para cada tipo de usuario, los cuales se muestran a continuación.

Manual de usuario para el tutor o padre de familia

Acceder a la siguiente dirección en un navegador web:

<http://cescolarnezatm.com/reinscripciones/>

Se mostrará la siguiente página, en donde se deberá escribir correctamente la clave CURP del alumno y enseguida hacer clic en el botón “BUSCAR ALUMNO”.

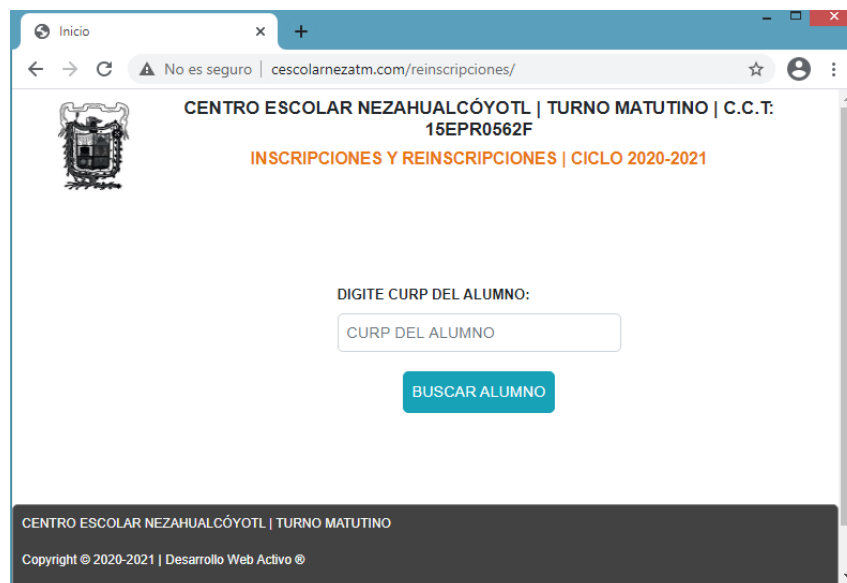


Figura A.1 Página de inicio para el tutor



Si la clave CURP no se encuentra, el sistema mostrará un mensaje que dice “NINGÚN RESULTADO”.

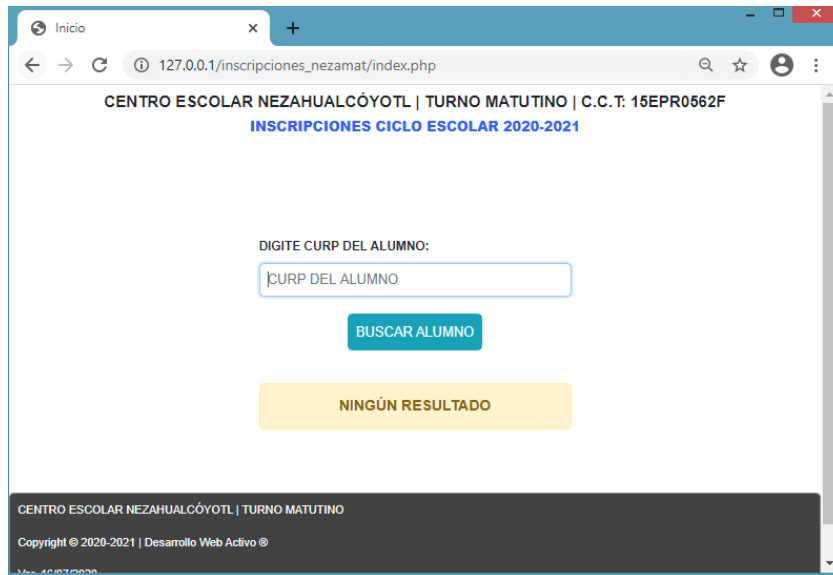


Figura A.2 Página de búsqueda de clave CURP

En caso de que el tutor haya realizado previamente el proceso de inscripción, el sistema mostrará junto con el nombre del alumno, un mensaje con la fecha y la hora en que el usuario realizó dicho proceso y se ofrecerá nuevamente la opción de imprimir el comprobante.



Figura A.3 Página con resultado de búsqueda

En caso contrario, el sistema mostrará el nombre del alumno encontrado y también un botón para iniciar el proceso de captura de datos que dice “INICIAR PROCESO DE INSCRIPCIÓN”.

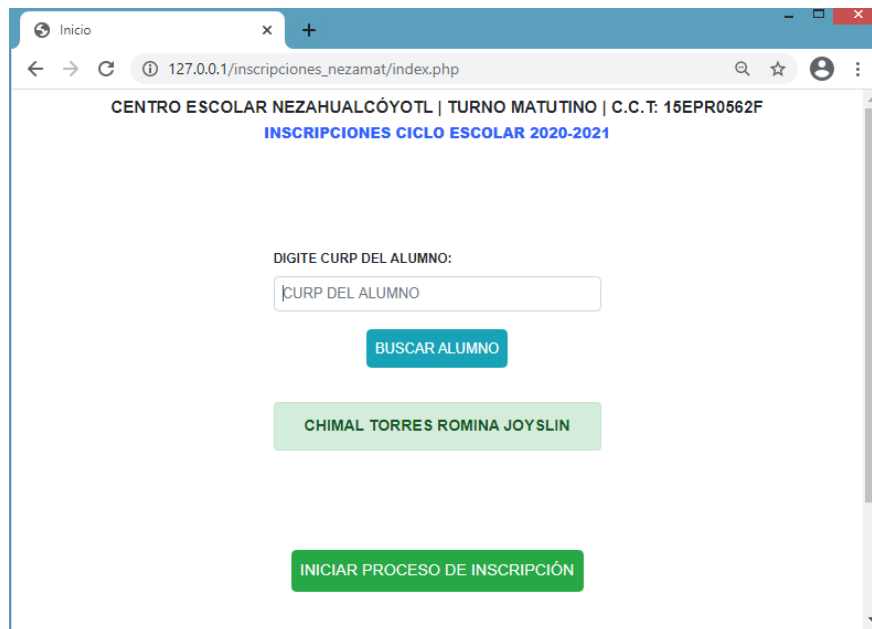


Figura A.4 Iniciar proceso de inscripción



Enseguida se mostrará una nueva página mostrando un formulario para ingresar datos del alumno y tutor. Llenar cuidadosamente cada campo.

* GRADO AL QUE SE INSCRIBIRÁ:

2°

* REPROBADO: NO SI

DATOS DE NACIMIENTO DEL ALUMNO

CURP DEL ALUMNO:

MAAF130428MMCRLRA0

NOMBRE DEL ALUMNO:

MARTINEZ ALVARADO MARIA FERNANDA

SEXO: M

FECHA DE NACIMIENTO: 2013-04-28

* ENTIDAD DE NACIMIENTO:

MEXICO

NACIONALIDAD: MEXICANA

DOMICILIO DEL ALUMNO

ESTADO: MEXICO

* MUNICIPIO:

Figura A.5 Ejemplo de llenado de formulario para el tutor



Al terminar de llenar el formulario hacer clic en el botón “TERMINAR Y ENVIAR”. Se sugiere revisar nuevamente todo el formulario para asegurarse de que no quedó ningún campo vacío y que los datos sean los correctos.

Figura A.6 Envío del formulario

Finalmente, el sistema muestra un aviso para que el usuario sepa que el alumno ha sido registrado con éxito y también pueda imprimir un comprobante que contiene un resumen de los datos ingresados.

INSCRIPCIONES CICLO ESCOLAR 2020-2021

Figura A.7 Botón imprimir comprobante



Ejemplo de la página que muestra un comprobante de inscripción para imprimir.

Comprobante de inscripción

127.0.0.1/inscripciones_benito/comprobante.php

DATOS DE SALUD DEL ALUMNO									
ESTATURA (cm):	75	PESO (Kg) :	50	TIPO DE SANGRE:	NR	ZAPATO ORTOPÉDICO:	NO	LENTES:	NO
ALERGIA:	NINGUNA			DISCAPACIDAD:	NINGUNA				
ENFERMEDADES QUE REQUIEREN ATENCIÓN ESPECIAL:									NINGUNA
VACUNAS									
COMPLETAS HASTA LOS 3 AÑOS:	NO	COMPLETAS HASTA LOS 6 AÑOS:	NO	COMPLETAS HASTA LOS 11 AÑOS:	NO	VACUNAS INCOMPLETAS:	NO		
HERRAMIENTAS DE APOYO:							NINGUNA		
DATOS DEL TUTOR									
CURP:	DURS090829HMCRDNA7			NOMBRE DEL PADRE O TUTOR:	LEON MARTINEZ MARIA				
SEXO:	M	FECHA DE NACIMIENTO:	1985-04-19	ENTIDAD DE NACIMIENTO:	MEXICO	NACIONALIDAD:	MEXICANA		
OCUPACIÓN:	EMPLEADO	ESTUDIOS:	BACHILLERATO TECNICO			PARENTESCO:	MADRE		
INGRESOS:	DE 2 A 3 SALARIOS MÍNIMOS				ESTADO CIVIL:	CASADA			
DATOS DE CONTACTO DEL PADRE O TUTOR									
TEL:	5546798255	TEL. EMER.	5513467985	E-MAIL:	PEDRO@GMAIL.COM	RED SOCIAL:	INSTAGRAM	USUARIO:	ASDFA
SOLICITANTE:					RESPONSABLE DE LA INSCRIPCIÓN:				
LEON MARTINEZ MARIA					SILVIA MARTÍNEZ OLIVA				

Fecha y hora de registro: 2020-07-25 20:39:44

Figura A.8 Ejemplo un comprobante de inscripción

Manual de usuario para el docente

Acceder al sistema ingresando a la siguiente dirección:

<http://cescolarnezatm.com/maestros/>

Escribir el nombre de usuario y contraseña que la dirección escolar proporcionó previamente.

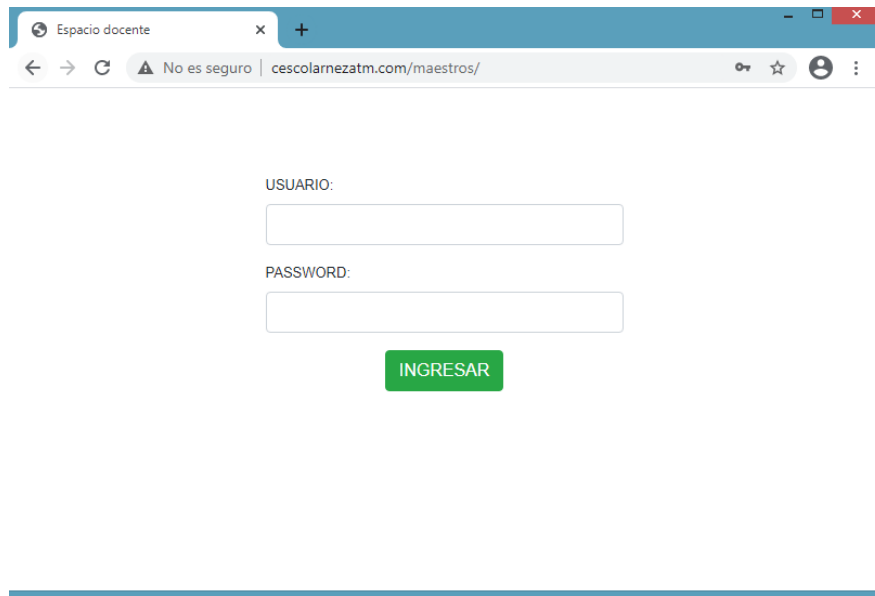


Figura A.9 Página de inicio para el docente

Una vez dentro del sistema el docente puede realizar las siguientes acciones:

- Consultar datos personales de algún alumno o tutor.
- Consultar calificaciones del grupo a su cargo.
- Capturar calificaciones trimestrales.

Consultar datos personales de alumnos y/o tutores.

Seleccionar en el menú la opción “Consulta grupo”, y después hacer clic en el grupo correspondiente. Se mostrará una lista de los alumnos del grupo a su cargo.

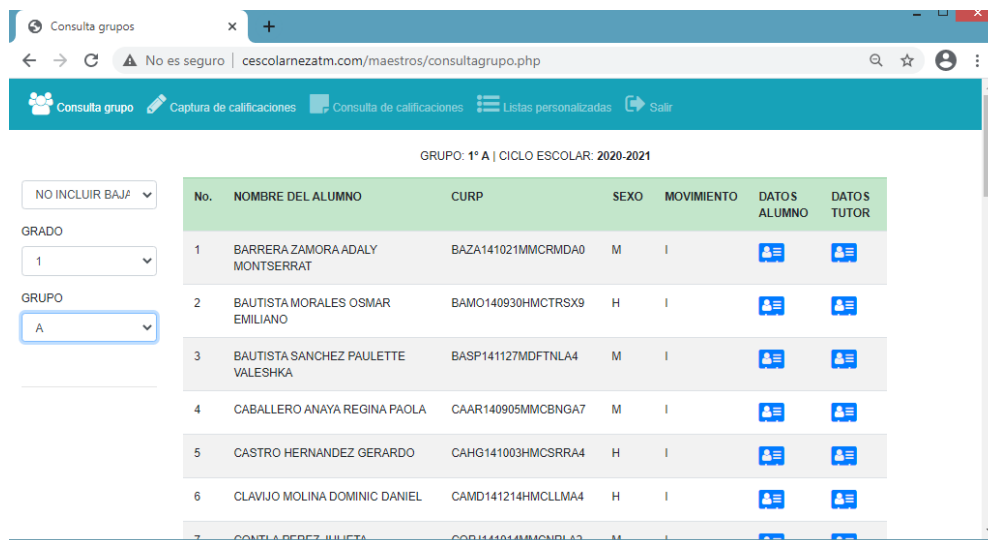




Figura A.10 Página de consulta de datos de alumnos y tutores

Se muestran dos botones al final de cada registro; el primero sirve para acceder a los datos del alumno y el segundo para los datos del tutor. Hacer clic en cualquiera de los dos botones según sea el caso para visualizar los datos deseados.

No.	NOMBRE DEL ALUMNO	CURP	SEXO	MOVIMIENTO	DATOS ALUMNO	DATOS TUTOR
1	BARRERA ZAMORA ADALY MONTSERRAT	BAZA141021MMCRMDA0	M	I		
2	BAUTISTA MORALES OSMAR EMILIANO	BAMO140930HMCTRSX9	H	I		
3	BAUTISTA SANCHEZ PAULETTE VALESHKA	BASP141127MDFTNLA4	M	I		
4	CABALLERO ANAYA REGINA PAOLA	CAAR140905MMCBNGA7	M	I		
5	CASTRO HERNANDEZ GERARDO	CAHG141003HMCSRRA4	H	I		
6	CLAVIJO MOLINA DOMINIC DANIEL	CAMD141214HMCLLMA4	H	I		
7	CONTRERAS JULIETA	CONR140914MMGNPLA3	M	I		

Figura A.11 Botones para acceso a datos de alumno y tutor



Se abrirá una nueva página la cual mostrará los datos completos del alumno o tutor, según hayamos seleccionado.

Consulta grupos x Datos del alumno x +

No es seguro | cescolarnezatm.c... 🔍 ☆ 👤 ⋮

ALUMN@: BARRERA ZAMORA ADALY MONTSERRAT

ID:	502
GRADO:	1
GRUPO:	A
MOVIMIENTO:	I
REPROBADO:	NO

DATOS DE NACIMIENTO

NOMBRE:	BARRERA ZAMORA ADALY MONTSERRAT
CURP:	BAZA141021MMCRMDA0
SEXO:	MUJER
FECHA DE NACIMIENTO:	2014-10-21

Figura A.12 Página con información de un alumno

Después de consultar los datos, es posible salir del sistema haciendo clic en el menú “Salir”.



Figura A.13 Salir del sistema

Consultar calificaciones del grupo a su cargo.

Una vez dentro del sistema seleccionar el menú “Consulta calificaciones”, enseguida elegir un periodo y una asignatura, después clic en el botón “Consultar”.



Figura A.14 Página de consulta de calificaciones

Se mostrará una lista de alumnos con sus calificaciones correspondientes al periodo y asignatura seleccionados.

Capturar o modificar calificaciones trimestrales.

Para realizar la captura o actualización de calificaciones trimestrales, ya dentro del sistema dirigirse al menú “Captura de calificaciones”. En caso de que la dirección escolar aún no haya habilitado la captura de algún periodo, el sistema mostrará el mensaje de “NO HAY HABILITADO NINGÚN PERIODO DE CAPTURA”.



Figura A.15 Captura de calificaciones no habilitada

Cuando ya existe un periodo de calificaciones previamente habilitado por la dirección escolar, entonces se podrá elegir una asignatura y se podrá acceder al proceso haciendo clic en el botón “CAPTURAR / MODIFICAR”.

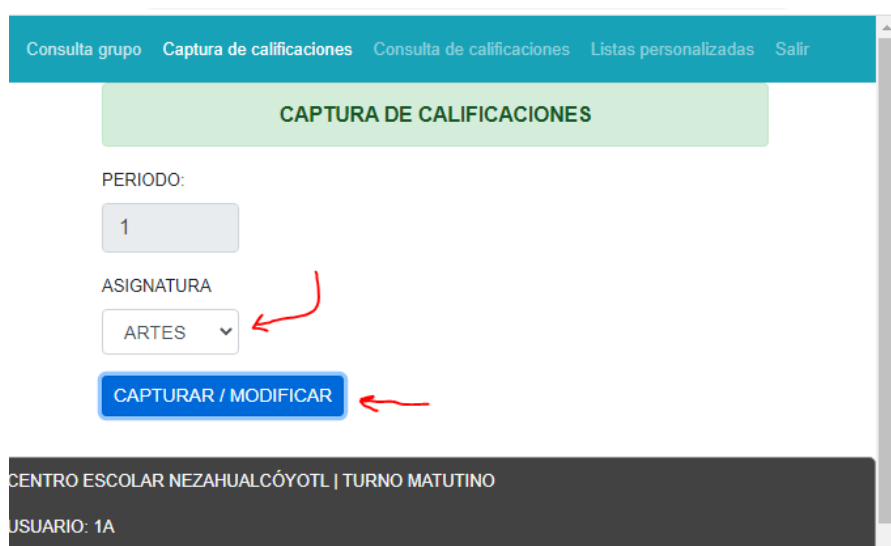


Figura A.16 Botón para actualizar calificaciones



Se mostrará una lista de los alumnos a su cargo con los campos para introducir las calificaciones correspondientes a cada alumno.

Captura de pantalla de la página web 'cescolarnezatm.com/maestros/update_calif.php'. La interfaz muestra un botón 'CAPTURAR / MODIFICAR' y un encabezado 'PERIODO: 2 | ARTES'. A continuación, se presenta una tabla con 9 filas de alumnos, cada una con un campo de entrada para la calificación.

No.	NOMBRE DEL ALUMNO	CALIFICACIÓN
1	BARRERA ZAMORA ADALY MONTSERRAT	<input type="text"/>
2	BAUTISTA MORALES OSMAR EMILIANO	<input type="text"/>
3	BAUTISTA SANCHEZ PAULETTE VALESHKA	<input type="text"/>
4	CABALLERO ANAYA REGINA PAOLA	<input type="text"/>
5	CASTRO HERNANDEZ GERARDO	<input type="text"/>
6	CLAVIJO MOLINA DOMINIC DANIEL	<input type="text"/>
7	CONTLA PEREZ JULIETA	<input type="text"/>
8	CUELLAR URIBE LAILA ODETTE	<input type="text"/>
9	DEHEZA VELAZQUEZ MELISSA ABIGAIL	<input type="text"/>

Figura A.17 Página de captura de calificaciones



Cuando se hayan terminado de capturar todas las calificaciones de la asignatura seleccionada se debe hacer clic en el botón “GUARDAR CAMBIOS” que aparece al final de la lista.

Captura de pantalla de un navegador web que muestra una lista de maestros con sus calificaciones. El navegador muestra la URL `cescolarnezatm.com/maestros/update_calif.php`. La lista contiene los siguientes datos:

ID	Nombre del Maestro	Calificación
27	RUIZ GARCIA YOLLOTZIN OCELOTL	8
28	SANCHEZ TORRES JOSE SEBASTIAN	8
29	SANDOVAL BASILIO FRIDA DANIELA	9
30	SANDOVAL ROSAS ATZIN DAYANNA	10
31	SANTIAGO ANIDES ALDO DAVID	8
32	SOLIS SANCHEZ IAN TADEO	9
33	SORIANO DE LA ROSA SANDRA CAMILA	10
34	TEMOSIHUE CORTES DIEGO	8
35	TORRES LARA SEBASTIAN	9

Debajo de la lista, hay un botón verde que dice "GUARDAR CAMBIOS", al cual apunta una flecha roja.

Centro Escolar Nezahualcōyotl | Turno Matutino
USUARIO: 1A
Copyright © 2020-2021 | Desarrollo Web Activo ©

Figura A.18 Botón guardar cambios de calificaciones



Manual de usuario para el administrador

Acceder al sistema ingresando a la siguiente dirección:

http://cescolarnezatm.com/control_escolar/

Escribir el nombre de usuario y contraseña correctos.

Control escolar

No es seguro | cescolarnezatm.com/control_escolar/

USUARIO:

PASSWORD:

INGRESAR

Figura A.19 Página de inicio del administrador

Una vez dentro del sistema el administrador puede realizar las siguientes acciones:

- Consultar datos personales de alumnos y tutores.
- Editar o actualizar datos de alumnos y tutores.
- Consultar calificaciones de cualquier grupo.
- Ingresar un nuevo alumno al sistema.
- Realizar búsquedas de alumnos.
- Eliminar alumnos del sistema.



Consultar datos personales de alumnos o tutores.

Seleccionar en el menú la opción “Grupos”. Seleccionar el grado y el grupo deseados de los controles de selección simple de la columna izquierda.

The screenshot shows a web browser window with the URL 'cescolarnezatm.com/control_escolar/consultagrupo.php'. The page title is 'Consulta grupos'. The navigation bar includes options like 'Búsqueda', 'Nuevo ingreso', 'Asignar grupos', 'Grupos', 'Calificaciones', '911', 'Periodo de calificaciones', 'Listas personalizadas', and 'Salir'. The main content area displays 'GRUPO: 1° A | CICLO ESCOLAR: 2020-2021'. On the left, there are filters for 'NO INCLUIR BA', 'GRADO' (set to 1°), and 'GRUPO' (set to A). The main table lists 6 students with columns for 'No.', 'NOMBRE DEL ALUMNO', 'CURP', 'SEXO', 'MOV', 'ALUMNO', and 'TUTOR'. Each row has icons for editing or viewing details for the student and tutor.

No.	NOMBRE DEL ALUMNO	CURP	SEXO	MOV	ALUMNO	TUTOR
1	BARRERA ZAMORA ADALY MONTSERRAT	BAZA141021MMCRMDA0	M	I	[Icons]	[Icons]
2	BAUTISTA MORALES OSMAR EMILIANO	BAMO140930HMCTRSX9	H	I	[Icons]	[Icons]
3	BAUTISTA SANCHEZ PAULETTE VALESHKA	BASP141127MDFTNLA4	M	I	[Icons]	[Icons]
4	CABALLERO ANAYA REGINA PAOLA	CAAR140905MMCBNGA7	M	I	[Icons]	[Icons]
5	CASTRO HERNANDEZ GERARDO	CAHG141003HMCSRRA4	H	I	[Icons]	[Icons]
6	CLAVIJO MOLINA DOMINIC DANIEL	CAMD141214HMCLLMA4	H	I	[Icons]	[Icons]

Figura A.20 Consulta de datos de alumnos y tutores

Hacer clic en algunos de los botones que representan los datos del alumno o tutor en las columnas “ALUMNO” o “TUTOR”. Se abrirá una página para visualizar los datos.

The screenshot shows two sections of personal data. The first section, 'DATOS DE NACIMIENTO', contains the following information: NOMBRE: BARRERA ZAMORA ADALY MONTSERRAT, CURP: BAZA141021MMCRMDA0, SEXO: MUJER, FECHA DE NACIMIENTO: 2014-10-21, ENTIDAD DE NACIMIENTO: MEXICO, NACIONALIDAD: MEXICANA. The second section, 'DOMICILIO', contains: ENTIDAD: MEXICO.

Figura A.21 Ejemplo de consulta de datos de un alumno

Editar o actualizar datos personales de alumnos o tutores.



Seleccionar en el menú la opción “Grupos”.

Seleccionar el grado y el grupo deseados de los controles de selección simple de la columna izquierda.

GRUPO: 1° A | CICLO ESCOLAR: 2020-2021

No.	NOMBRE DEL ALUMNO	CURP	SEXO	MOV	ALUMNO	TUTOR
1	BARRERA ZAMORA ADALY MONTSERRAT	BAZA141021MMCRMDA0	M	I	 	 
2	BAUTISTA MORALES OSMAR EMILIANO	BAMO140930HMCTRSX9	H	I	 	 
3	BAUTISTA SANCHEZ PAULETTE VALESHKA	BASP141127MDFTNLA4	M	I	 	 
4	CABALLERO ANAYA REGINA PAOLA	CAAR140905MMCBNGA7	M	I	 	 

Figura A.22 Botones de actualización de datos de alumnos y tutores

Hacer clic en algunos de los botones que representan la edición de los datos del alumno o tutor (los íconos que tienen un lápiz) en las columnas “ALUMNO” o “TUTOR”. Se abrirá un formulario de edición para actualizar los datos.

* ENTRE CALLE:

* Y CALLE:

* NÚMERO EXTERIOR:

NÚMERO INTERIOR:

* REFERENCIA:


ACTUALIZAR DATOS 

Figura A.23 Formulario de edición de datos

Al terminar de editar los campos necesarios, se debe hacer clic en el botón “ACTUALIZAR DATOS”.



Consultar calificaciones de cualquier grupo

Seleccionar la opción “Calificaciones” del menú principal.

En la columna de la izquierda seleccionar el periodo, el grado y el grupo deseados y hacer clic en el botón “Consultar”. Se podrán visualizar las calificaciones de todas las asignaturas de cada uno de los alumnos del grupo seleccionado.

PERIODO: 1 GRUPO: 1º A | CICLO ESCOLAR: 2020-2021

No.	NOMBRE DEL ALUMNO	L. M. ESP	MATEMÁTICAS	C. DEL MEDIO	F. C. Y ÉTICA	ARTES	E. FÍSICA	INASISTENCIAS
1	BARRERA ZAMORA ADALY MONTSERRAT	10	10	10		10	10	0
2	BAUTISTA MORALES OSMAR EMILIANO	9	9	9		9	9	0
3	BAUTISTA SANCHEZ PAULETTE VALESHKA	10	10	10		10	10	0
4	CABALLERO ANAYA REGINA PAOLA	10	10	10		10	10	0
5	CASTRO HERNANDEZ	8	8	8		8	8	0

Figura A.24 Consulta de calificaciones por grupo



Ingresar un nuevo alumno al sistema.

Dentro del sistema dirigirse a la opción “Nuevo ingreso” del menú principal.

Ingresar en cada campo los datos correspondientes y después hacer clic en el botón “INGRESAR”. Una vez realizado dicho proceso se debe informar al padre de familia o tutor que ya puede ingresar sus datos en la página correspondiente.

La imagen muestra una interfaz web en un navegador. La barra de direcciones muestra la URL `cescolarnezatm.com/control_escolar/insertar_alumno.php`. El encabezado de la página incluye el texto "Nezahualcóyotl | Matutino" y un menú de navegación con íconos para: Búsqueda, Nuevo ingreso, Asignar grupos, Grupos, Calificaciones, 911, Periodo de calificaciones, Listas personalizadas y Salir. El contenido principal de la página está titulado "Alumno nuevo ingreso" y contiene un formulario con los siguientes campos:

- * CURP DEL ALUMNO:
- A. PATERNO:
- A. MATERNO:
- NOMBRE(S):

Debajo de los campos hay un botón verde con el texto "INGRESAR".

Figura A.25 Ingresar un nuevo alumno



Realizar una búsqueda de alumnos

Para realizar una búsqueda de alumnos en el sistema se debe ir a la opción “Búsqueda” del menú principal. Enseguida se debe escribir el nombre del alumno a buscar en el campo del formulario que dice “Buscar por nombre”.

Se mostrará una lista con las coincidencias encontradas, desde esta lista es también posible realizar la edición o la eliminación definitiva de algún alumno, utilizando los botones correspondientes de las últimas columnas (EDITAR ALUMNO, EDITAR TUTOR, ELIMINAR ALUMNO).

No.	NOMBRE DEL ALUMNO	GRADO	GRUPO	CURP	SEXO	MOVIMIENTO	EDITAR ALUMNO	EDITAR TUTOR	ELIMINAR ALUMNO
1	MERAZ CERVANTES KARLA SELINA	5	E	MECK101031MMCRRA1	M	I			
2	MERAZ CERVANTES ROUSS MERI	1	B	MECR140925MMCRSA3	M	I			

Figura A.26 Botones de edición y eliminación de registros

Bibliografía

Beynon - Davies, P (2014). *Sistemas de Información Introducción a la Informática en las organizaciones*. Barcelona, España: Editorial: Reverté.

Casado, C (2014). *Entornos de desarrollo*. Madrid, España: Editorial RA-MA.

Ferrer, J (2014). *Implantación de aplicaciones web*. Madrid, España: Editorial RA-MA.

Ferrer, J (2015). *Implantación de aplicaciones web en entornos internet, intranet y extranet*. Madrid, España: Editorial RA-MA.

Inee (2019). *Panorama educativo en México 2018, indicadores del sistema educativo nacional educación básica y media superior*. Disponible en: <https://www.inee.edu.mx/wp-content/uploads/2019/08/P1B117.pdf>

Juganaru, M (2014). *Introducción a la programación*. México: Editorial Patria.

Loic, G (2015). *Introducción a la ingeniería de software, modelo de desarrollo de programas*. Madrid, España: Delta publicaciones universitarias.

Marco, M (2010). *Escaneando la informática*. Barcelona, España: Editorial UOC.

Nevado, M (2015). *Introducción a las bases de datos relacionales*. Madrid, España: Editorial Visión Libros.

Osorio, F (2008). *Bases de datos relacionales teoría y práctica*. Medellín, Colombia: Instituto Tecnológico Metropolitano.

Pantaleo, G (2015). *Ingeniería de software*. México: Alfaomega grupo editor.

Piñeiro, J (2014). *Diseño de bases de datos relacionales*. España: Ediciones Paraninfo.

Ramos, A & Moreno, J (2014). *Gestión de servicios en el sistema informático*. Madrid, España: Editorial RA-MA.

Sommerville, I (2005). *Ingeniería del software*. Madrid, España: Pearson Educación

Unicef (s.f). *Educación básica e igualdad entre los géneros*. Disponible en https://www.unicef.org/spanish/education/index_bigpicture.html.