

FUNDACIÓN UNIVERSITARIA DEL ÁREA ANDINA

FACULTAD DE INGENIERÍA

INGENIERÍA DE SISTEMAS

PROYECTO DE GRADO

**DISEÑO Y DESARROLLO DE UNA APLICACIÓN WEB PARA LA GESTIÓN DE NOTAS EN LAS
ESCUELAS DEPORTIVAS DEL COLEGIO GIMNASIO CAMPESTRE**

CRISTIAN YECID ESPITIA BARRETO

Cód. 20171508211N



Bogotá, Colombia

2021

Tabla de contenido

Tabla de ilustraciones.....	4
Índice de tablas.....	5
Introducción	7
Identificación del problema.....	7
Formulación del Problema	7
Justificación	8
Objetivos.....	9
General	9
Específicos	9
Alcance.....	10
Metodología de Desarrollo.....	11
Metodología UWE	13
Fases de la metodología UWE	14
Fase de análisis	14
Listado de requerimientos funcionales	14
Listado de requerimientos no funcionales.....	27
Fase de diseño del sistema.....	28
Diagrama inicio de sesión.....	28
Diagrama de usuarios.....	29
Diagrama de estudiantes.....	30
Diagrama escuelas deportivas.....	31
Diagrama descriptores de logro	32
Diagrama calificación.....	33
Diagrama físico	34
Diagrama de Clases.....	35
Roles en la aplicación	36
Modelado de datos.....	36
Modelo Relacional	36

Diccionario de datos	38
Modelo de Navegación.....	42
Modelo de navegación perfil jefe de escuelas	44
Modelo de navegación perfil instructor	45
Modelo de navegación perfil sistemas.....	46
Modelo de presentación	47
Inicio de sesión	47
Opciones aplicación.....	48
Módulo escuelas.....	48
Registrar descriptor de logro.....	49
Calificación de estudiante	50
Envío de informe por correo electrónico	51
Inscripción de estudiante	52
Fase de codificación del software	52
HTML.....	53
PHP.....	54
JAVASCRIPT	55
CSS	56
Fase de pruebas.....	57
Prueba de acceso e inicio de aplicación web.	59
Prueba de inicio de sesión.....	60
Prueba de gestión de usuarios.	60
Prueba de gestión de las escuelas deportivas.....	61
Prueba de gestión descriptores de logro.	62
Prueba de gestión de estudiantes.....	63
Prueba de inscripción de un estudiante en escuela deportiva.	64
Prueba de calificación estudiante.	65
Fase de implementación	66
Fase mantenimiento y aseguramiento.....	69
Plan de calidad.....	70
Matriz de riesgos.....	72

Conclusiones.....	73
Bibliografía.....	74

Tabla de ilustraciones

Ilustración 1- Metodologías de desarrollo del Software.....	11
Ilustración 2- Logotipo UWE.....	13
Ilustración 3- Descripción general de los modelos UWE.....	13
ILUSTRACIÓN 4- DIAGRAMA INICIO SESIÓN.....	28
ILUSTRACIÓN 5- DIAGRAMA USUARIOS.....	29
Ilustración 6- Diagrama de estudiantes.....	30
Ilustración 7- Diagrama escuelas deportivas.....	31
Ilustración 8- Diagrama Descriptores de Logro.....	32
Ilustración 9- Diagrama Calificación.....	33
Ilustración 10- Diagrama físico.....	34
ILUSTRACIÓN 11- DIAGRAMA DE CLASES.....	35
Ilustración 12- Modelo relacional.....	37
Ilustración 13- Modelo navegación perfil Jefe de escuelas.....	44
ILUSTRACIÓN 14- MODELO NAVEGACIÓN PERFIL INSTRUCTOR.....	45
Ilustración 15- Modelo navegación perfil sistemas.....	46
Ilustración 16- Modelo presentación: Inicio se sesión.....	47
Ilustración 17- Modelo presentación opciones aplicación.....	48
Ilustración 18- Modelo presentación: modulo escuelas.....	48
Ilustración 19- Modelo presentación: Listado escuelas.....	49
ILUSTRACIÓN 20- MODELO PRESENTACIÓN: REGISTRAR DESCRIPTOR DE LOGRO.....	49
Ilustración 21- Modelo presentación: calificación de estudiante.....	50
Ilustración 22- Modelo presentación: calificación descriptores.....	50
Ilustración 23- Modelo presentación: envío de informe.....	51
Ilustración 24- Modelo presentación: inscripción de estudiante.....	52
Ilustración 25- Fragmento de código HTML.....	53
Ilustración 26- Fragmento de código PHP.....	54
Ilustración 27- Fragmento de código JavaScript.....	55
Ilustración 28- Fragmento de código CSS.....	56
Ilustración 29- Instalación servidor.....	57
Ilustración 30- Configuración máquina virtual.....	57
Ilustración 31- Instalación servicio servidor Web.....	58
Ilustración 32- Verificación de servicios Web.....	58
Ilustración 33- Instalación servidor Implementación.....	66

Ilustración 34- Configuración máquina virtual implementación.....	66
Ilustración 35- Servicios Web implementación.....	67
Ilustración 36- Verificación servidor Web implementación.....	67
Ilustración 37- Base de datos limpia.....	68
Ilustración 38- Estado aplicación web funcional.....	68
ILUSTRACIÓN 39- UBICACIÓN DE ARCHIVOS	69
Ilustración 40- Plan de calidad.....	70

Índice de tablas

Tabla 1- Requerimiento Funcional 001	16
Tabla 2- Requerimiento Funcional 002	17
Tabla 3- Requerimiento Funcional 003	18
Tabla 4- Requerimiento Funcional 004	20
Tabla 5- Requerimiento Funcional 005	21
Tabla 6- Requerimiento Funcional 006	22
Tabla 7- Requerimiento Funcional 007	23
Tabla 8- Requerimiento Funcional 008	24
Tabla 9- Requerimiento Funcional 009	26
Tabla 10- Requerimiento Funcional 010	26
Tabla 11- Requerimiento Funcional 011	27
Tabla 12- Roles de usuario	36
Tabla 13- Tabla ER: módulos	38
Tabla 14- Índices de modulos.....	38
Tabla 15- Tabla ER: aspectos	38
Tabla 16- Índices de aspectos.....	38
Tabla 17- Tabla ER: descriptores	39
Tabla 18- Índices de descriptores.....	39
Tabla 19- Tabla ER: escueladep	39
Tabla 20- Índices de escueladep.....	39
Tabla 21- Tabla ER: estudescriptores	39
Tabla 22- Índices de tabla estudescriptores.....	39
Tabla 23- Tabla ER: estudiantes.....	40
Tabla 24- Índices tabla estudiantes.....	40
Tabla 25- Tabla ER: inscripciones	40
Tabla 26- Índices de tabla inscripciones.....	40
Tabla 27- Tabla ER: menus	40
Tabla 28- Índices de tabla menus.....	41

Tabla 29- Tabla ER: modulos	41
Tabla 30- Índices de tabla modulos.....	41
Tabla 31- Tabla ER: usuarioescuela	41
Tabla 32- Índices de tabla usuarioescuela.....	41
Tabla 33- Tabla ER: usuarios.....	42
Tabla 34- Índices de tabla usuarios	42
Tabla 35 - Formato para pruebas del sistema.....	59
Tabla 36- Prueba: acceso e inicio de aplicación web	59
Tabla 37- Prueba: inicio de sesión	60
Tabla 38- Prueba: Gestión de usuarios.....	61
Tabla 39- Prueba: Gestión escuelas deportivas	62
Tabla 40- Prueba: gestión descriptores de logro	63
Tabla 41- Prueba: Gestión de estudiantes	64
Tabla 42- Prueba: inscripción de un estudiante.....	64
Tabla 43- Prueba: Calificación de un estudiante.....	65
Tabla 44- Descripción plan de calidad	71
Tabla 45- Matriz de riesgos	72

Introducción

En la actualidad, la tecnología como medio para lograr objetivos de aprendizaje ha tomado un mayor auge la cual resulta cada vez más necesaria. No solo facilita presentar información, sino entender métodos y ayudas para gestionar los aspectos pedagógicos de un ambiente escolar que cada día crece y se vuelve más enriquecido por su constante evolución.

A continuación, se tratará de aplicar las ayudas tecnológicas para la gestión de información de manera computarizada y económica. Lo que garantizará seguir procesos pedagógicos actuales generando pequeños cambios en los procesos garantizando la gestión de la información tratando de sustituir el uso inadecuado del papel impreso que no siempre es la mejor forma de guardar información.

Identificación del problema

Teniendo en cuenta que a través de los años en el colegio Gimnasio Campestre ha aumentado la participación de los estudiantes en las escuelas deportivas que se llevan a cabo diferentes horarios de lunes a sábados después de las jornadas académicas. Los instructores en algunos casos, llevan el registro de los logros y el control de sus estudiantes mediante archivos planos de (Word, Excel), dejando muchas veces información sin registrar; Por lo que no se podía brindar de manera adecuada y generalizada un reporte del progreso del estudiante. Otro punto a considerar es la seguridad de los datos sensibles de los estudiantes que son expuestos a libre manipulación por parte de instructores.

Dado el crecimiento de estudiantes inscritos en las escuelas deportivas que está en aumento y es muy difícil de maniobrar, era necesario tener una herramienta para el control de los estudiantes inscritos a cada escuela unificando los criterios de evaluación de las diferentes escuelas y así brindar el informe de la evolución del estudiante frente a los criterios que se deberían alcanzar.

Formulación del Problema

¿El desarrollo e implementación de un sistema de gestión de notas que permitirá al Colegio Gimnasio Campestre mejorar sus procesos de registro y verificar los logros alcanzados por los estudiantes en las escuelas deportivas?

Justificación

A través de los años el Colegio Gimnasio Campestre ha impulsado la participación de los estudiantes en las escuelas de formación en diferentes áreas deportivas, artística y del conocimiento en diferentes horarios de la jornada de la tarde, después de la jornada académica e incluso los sábados en la mañana, las actividades se desarrolla de acuerdo a la edad de sus participantes, motivo por el que algunas escuelas se dividen en categorías, con el fin de garantizar que los propósitos que persiguen tengan concordancia con el grupo al que está dirigido el programa y con el desarrollo personal y escolar de sus participantes, en la cual trabaja tres dimensiones: formación deportiva, formación artística y de conocimiento. (Colegio Gimnasio Campestre, 2019)

Debido a la gran acogida de estudiantes por las escuelas deportivas se ve la necesidad de centralizar la información de las escuelas como: descriptores de logros, estudiantes inscritos en cada escuela, calificaciones generadas y más datos relevantes que no se encuentran en un solo repositorio. Es necesario una optimización de procesos sin dejar de asegurar el cumplimiento de las escuelas deportivas se hace necesario crear una aplicación web para la gestión de calificaciones de estudiantes que participan en las escuelas deportivas del Colegio Gimnasio Campestre donde los instructores podrán calificar, generar y enviar los informes por correo electrónico a los acudientes de los estudiantes inscritos en la escuela designada.

Para lo cual los principales pilares de este proyecto será reducir el uso de papel y evitar el registro en aplicaciones como “Microsoft Word” o “Microsoft Excel”, ya que no siempre son la mejor forma de gestionar la información por lo que no es posible acceder a todos los archivos y datos creados de manera ágil e inmediata.

Objetivos

General

- Diseñar e implementar una aplicación web segura, para gestionar la información del registro, control y seguimiento de los estudiantes semestralmente en las escuelas deportivas del Gimnasio Campestre para su posterior informe.

Específicos

- Recolectar información sobre el funcionamiento de las escuelas deportivas para identificar los requerimientos funcionales y no funcionales presentes en la gestión de los datos.
- Elaborar un repositorio de datos referente a las escuelas deportivas del Gimnasio Campestre acorde a los requerimientos de seguridad para su control y gestión.
- Plantear el algoritmo y diseño que permita evaluar a los estudiantes por parte del instructor y posteriormente él envíe automáticamente de los informes generados por escuela a los correos designados.
- Gestionar los datos personales de los estudiantes controlando el acceso a los instructores.

Alcance

Se desarrollará una aplicación web funcional y acorde con los lineamientos de seguridad, confiabilidad y calidad, la cual será instalada en los servidores del colegio Gimnasio Campestre, para que pueda ser utilizada por los instructores y demás personal delegado para su uso, en consecuencia, no se generarán costos adicionales en la operación y brinda una herramienta tecnológica para que los instructores tengan centralizado toda la información referente a las calificaciones de las escuelas deportivas.

Unificar el medio de transmisión de información para generar el reporte del estudiante en un formato claro para que esté llegue al acudiente vía correo electrónico sin perder información en el proceso. La finalidad de la aplicación será enviar vía correo electrónico adjuntando los informes finales por semestre de cada estudiante inscrito en una escuela de formación, otros informes que se tienen desde la oficina de escuelas no serán generados por la aplicación las cuales pueden ser retomadas en mejoras futuras.

Se entregarán manuales de usuario por perfil donde se les explicará el funcionamiento de la aplicación, con sus respectivas validaciones con el fin de tener un instrumento de consulta unificada para ser consultada cuando se requiera por parte de los empleados del Gimnasio Campestre.

El código fuente tendrán su documentación respectiva en funciones de lógica de negocio, esto indica que no se hará manual de código fuente para desarrollador. Por otro lado, luego de la instalación, implementación y entrega de la aplicación no se realizarán ajustes de código, para esto se entregará el código fuente de la aplicación para posteriores cambios, correcciones o adiciones de nuevas funcionalidades.

Metodología de Desarrollo

Una metodología de desarrollo de software se refiere a un marco de trabajo que es usado para estructurar, planear y controlar el proceso de desarrollo en sistemas de información con el objetivo principal de aumentar la calidad del software que se produce en todas y cada una de sus fases de desarrollo (EcuRed, 2019)

En el desarrollo de aplicaciones existe gran cantidad de metodologías, por lo cual, no existe una universal, porque cada metodología debe adaptarse al proyecto en particular teniendo en cuenta los recursos técnicos, recurso humano, etc.



Ilustración 1- Metodologías de desarrollo del Software.

FUENTE: [HTTPS://WWW.AMERICANINDUSTRIALMAGAZINE.COM/BLOGS/NEGOCIOS/LINKERS-CREANDO-ENGAGEMENT-ENTRE-LAS-ORGANIZACIONES-Y-COLABORADORES](https://www.americanindustrialmagazine.com/blogs/negocios/linkers-creando-engagement-entre-las-organizaciones-y-colaboradores)

Algunas de las metodologías más usadas son:

- **Desarrollo de sistemas de Jackson (JSD):** La metodología de Jackson a partir del trabajo de M.A. Jackson, sobre el análisis del dominio de la información y sus relaciones con el diseño de programas y sistemas, utilizando la técnica del diseño descendente, cuyo producto final es el programa escrito, de tal forma que su codificación posterior en cualquier lenguaje de programación es relativamente sencilla (edukativos, 2016)
- **Rapid Application Development (Desarrollo rápido de aplicaciones – RAD):** Es un modelo de desarrollo de software, que implica el desarrollo iterativo y la

construcción de prototipos. El desarrollo rápido de aplicaciones es un término originalmente utilizado para describir un proceso de desarrollo de software introducido por James Martin en 1991. Si se perciben bien los requisitos y se limita el entorno del proyecto permite al equipo de desarrollo crear un sistema funcional dentro de períodos cortos de tiempo. (esacademic.com, 2010)

- **Metodología Scrum:** Es una estructura de procesos ágiles de desarrollo el cual usa una serie de procesos en el que se aplican de manera regular un conjunto de buenas prácticas para trabajar colaborativamente, en equipo, y obtener el mejor resultado posible de un proyecto que requiere rapidez y flexibilidad. (proyectosagiles.org, 2008)
- **Metodología Extreme programming. (Programación extrema – XP):** Es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. En la que se da máxima prioridad a la obtención de resultados y reduce la burocracia que se produce al utilizar otras metodologías. (EcuRed, 2019)
- **Metodología UWE:** es un método de ingeniería del software para el dominio web cuyo objetivo es cubrir todo el ciclo de vida del desarrollo de aplicaciones web basado en UML. Cualquier tipo de diagrama UML puede ser usado, porque UWE es una extensión de UML. El enfoque principal del enfoque UWE es proporcionar un lenguaje de modelado específico de dominio basado en UML que también incluye características de seguridad, metodología basada en modelos, soporte de herramientas para el diseño sistemático, y soporte de herramientas para la generación semiautomática de aplicaciones web. (Nora Koch, Marianne Busch, 2016)

Esta última metodología fue escogida por lo que está enfocada sobre el diseño, la personalización y la generación de escenarios que guían al proceso de desarrollo de aplicaciones web, la cual se basa en técnicas de UML¹.

El aplicativo Web en este proyecto, se basa en las necesidades que tienen en el colegio específicamente las escuelas deportivas en general, en cuanto al desarrollo y diseño del aplicativo Web, dando como prioridad a los requerimientos que los usuarios requieren y desean, ya que son ellos quienes utilizaran dicho aplicativo.

¹ El lenguaje unificado de modelado (UML, por sus siglas en inglés, Unified Modeling Language) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad.

Metodología UWE

UWE es un proceso de desarrollo para aplicaciones Web enfocado sobre el diseño sistemático, la personalización y la generación semiautomática de escenarios que guíen el proceso de desarrollo de una aplicación Web. UWE describe una metodología de diseño sistemática, basada en las técnicas de UML, la notación de UML y los mecanismos de extensión de UML. UWE es una propuesta basada en el proceso unificado y UML, pero adaptados a la web. En requisitos separa las fases de captura, definición y validación. Hace además una clasificación y un tratamiento especial dependiendo del carácter de cada requisito.



Ilustración 2- Logotipo UWE

FUENTE: [HTTP://UWE.PST.IFI.LMU.DE/_LAYOUT/UWELOGOMAIN.PNG](http://uwe.pst.ifi.lmu.de/_LAYOUT/UWELOGOMAIN.PNG)

El método UWE consiste en la construcción de modelos de análisis y diseño, dicha construcción se realiza dentro del marco de un proceso de diseño iterativo e incremental. Las actividades de modelado abarcan: el análisis de requerimientos, diseño conceptual, modelo de usuario, diseño de la navegación, de la presentación y diseño de la adaptación.

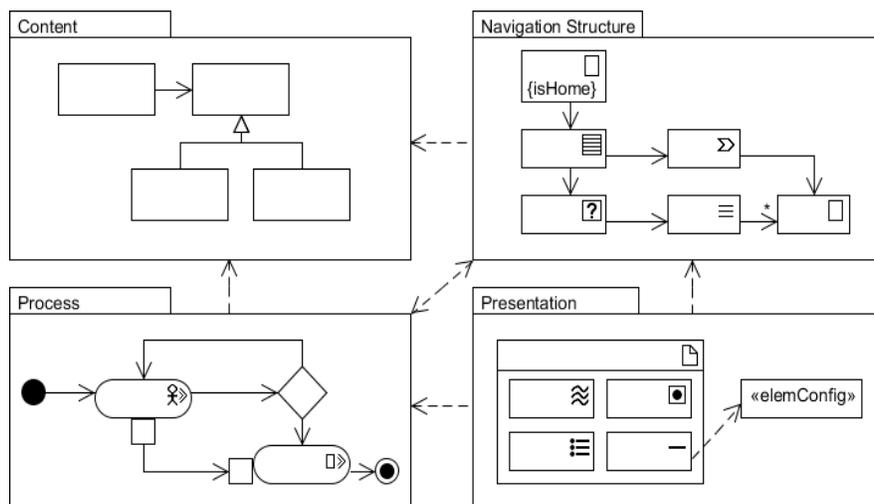


Ilustración 3- Descripción general de los modelos UWE

Fuente: <http://uwe.pst.ifi.lmu.de/examples/UWE-overview.png>

Fases de la metodología UWE

- **Fase de análisis:** Captura, análisis y especificación de requisitos durante esta fase, se adquieren, reúnen y especifican las características funcionales y no funcionales que deberá cumplir la aplicación web.
- **Fase de diseño del sistema:** Se basa en la especificación de requisitos producto del análisis de los requerimientos (fase de análisis), el diseño define cómo estos requisitos se cumplirán, la estructura que debe darse a la aplicación web. Los diagramas a tenerse en cuenta son:
 - Diagrama de Casos de Usos
 - Diagrama Físico
 - Diagrama de Clases
 - Modelo Navegación
 - Modelo de Presentación
- **Fase de codificación del software:** Durante esta etapa se realizan las tareas de programación; que consiste, esencialmente, en llevar a código fuente, al lenguaje de programación elegido, todo lo diseñado en la fase anterior.
- **Fase de pruebas:** en este punto se asegura el correcto funcionamiento de parte del programa o secciones de código para su verificación y validación.
- **Fase de implementación:** en esta fase, los códigos fuente desarrollados se llevan al servidor eventualmente configurado, con el propósito de ser utilizado por los usuarios finales. Es importante utilizar estándares para asegurar la futura compatibilidad y escalabilidad del aplicativo.
- **Fase mantenimiento y aseguramiento:** este proceso de control, mejora y optimización del aplicativo ya desarrollado, el cual incluye depuración de errores y defectos que puedan haberse filtrado de la fase de pruebas de control. Los problemas de uso no detectados durante el proceso de desarrollo pueden descubrirse a través de varios métodos, principalmente a través de los mensajes, opiniones de los usuarios, el comportamiento y uso del sitio.

Fase de análisis

El primer paso para el desarrollo del aplicativo web que se especificará con UWE, es realizar la identificación de los requerimientos para luego plasmarlos en un modelo donde permita clasificarlos en requerimientos funcionales (RF00) y requerimientos no funcionales(RNF00).

Listado de requerimientos funcionales

Los requerimientos funcionales son afirmaciones de los servicios que realizará el aplicativo los cuales, debe estar completos y ser consistentes.

A continuación, enumeramos los requerimientos que fueron obtenidos:

- RF001: Los usuarios deberán identificarse y validar sus permisos en el sistema.
- RF002: En el sistema en un módulo debe gestionar el registro de usuarios para el acceso al sistema y poder navegar en él según el perfil asociado, de forma queden encriptadas sus credenciales.
- RF003: El sistema debe permitir gestionar los datos básicos de las escuelas deportivas que se vayan a asignar.
- RF004: El sistema debe gestionar los datos de los descriptores de logros y asociarlos adecuadamente a las escuelas registradas en el sistema.
- RF005: El sistema debe gestionar los datos básicos de los estudiantes para el posterior envío de informes.
- RF006: En el sistema se deben poder inscribir los estudiantes registrados en más de una escuela.
- RF007: El sistema debe admitir la calificación de los estudiantes inscritos en una escuela y listar los que faltan por calificar.
- RF008: El sistema debe admitir la actualización de la calificación realizada de los estudiantes inscritos en una escuela.
- RF009: El sistema debe generar un reporte de los estudiantes con calificación y poder visualizar la respectiva calificación.
- RF010: Cuando se termine de actualizar los datos del estudiante escogido, transferir estos datos aún formato PDF para enviar por correo electrónico.
- RF011: El sistema debe registrar el usuario que realice cambios o actualizaciones de la información registrada.

DEFINICIÓN DEL REQUERIMIENTO					
NO. REQUERIMIENTO		RF001	Los usuarios deberán identificarse y validar sus permisos en el sistema.		
PRIORIDAD	Media	IMPORTANCIA	Alta	FECHA ELABORACIÓN	29/09/2020
RESPONSABLES		CRISTIAN ESPITIA			
DESCRIPCIÓN		<p>El sistema en su página de ingreso debe verificar que el usuario que está intentando ingresar esté registrado y sus credenciales sean correctas.</p> <p>Debe solicitar los datos de nombre de usuario(alfanumérico) y contraseña (alfanumérica).</p> <p>Al momento de ingresar al sistema, se deben buscar y asignar los accesos a los módulos y formularios a los que tiene permiso según el perfil asignado desde la creación del usuario.</p>			

Asociar en una variable el usuario para registrar los cambios que el usuario realice en el sistema.			
REQUERIMIENTO ASOCIADO		FUNCIONAL	REQUERIMIENTO NO FUNCIONAL ASOCIADO
<ul style="list-style-type: none"> ● RF002 			<ul style="list-style-type: none"> ▪ RNF002 ▪ RNF003 ▪ RNF004 ▪ RNF006

Tabla 1- Requerimiento Funcional 001

DEFINICIÓN DEL REQUERIMIENTO					
NO. REQUERIMIENTO		RF002	En el sistema en un módulo debe gestionar el registro de usuarios para el acceso al sistema y poder navegar en él según el perfil asociado.		
PRIORIDAD	Media	IMPORTANCIA	Media	FECHA ELABORACIÓN	29/09/2020
RESPONSABLES		CRISTIAN ESPITIA			
DESCRIPCIÓN		<p>El sistema debe tener un módulo con formularios web para registrar, actualizar, listar usuarios y la posibilidad de cambiar la clave del usuario que está en la aplicación.</p> <p>Registrar usuario: en un formulario se debe poder ingresar un nuevo usuario debe aparecer tanto la fecha en un campo visible como el nombre de usuario de quien está creando el nuevo usuario; los datos que se deben solicitar para actualizar son:</p> <ul style="list-style-type: none"> ● Nombre de usuario ● Clave para entregar al usuario ● Nombres del usuario ● Seleccionar de una lista desplegable un perfil de permisos ● Seleccionar una escuela asociada. <p>Tener un botón para guardar y que genere un mensaje de la creación correcta.</p> <p>Actualizar Usuario: en un formulario se debe poder actualizar un usuario se debe seleccionar el usuario desde una lista desplegable y también debe aparecer tanto la fecha en un capo</p>			

	<p>visible como el nombre de usuario de quien está actualizando el usuario; los datos que se deben solicitar son:</p> <ul style="list-style-type: none"> ● Nombre de usuario ● Clave para entregar al usuario ● Nombres del usuario ● Seleccionar de una lista desplegable un perfil de permisos ● Seleccionar una escuela asociada. ● Seleccionar el estado del usuario (activo, inactivo) <p>Tener un botón para actualizar y que genere un mensaje de la actualización correcta.</p> <p>Listado de usuarios: en un formulario se debe poder listar los usuarios registrados en el sistema desde una lista desplegable por perfil o todos. Los resultados deben mostrarse en una tabla ordenada mostrando nombres, usuario, escuela deportiva, y estado.</p>		
	REQUERIMIENTO ASOCIADO	FUNCIONAL	REQUERIMIENTO NO FUNCIONAL ASOCIADO
			<ul style="list-style-type: none"> ▪ RNF002 ▪ RNF003 ▪ RNF004 ▪ RNF006 ▪ RNF009

Tabla 2- Requerimiento Funcional 002

DEFINICIÓN DEL REQUERIMIENTO					
NO. REQUERIMIENTO		RF003	El sistema debe gestionar los datos básicos de las escuelas deportivas que se vayan a asignar.		
PRIORIDAD	Media	IMPORTANCIA	Media	FECHA ELABORACIÓN	29/09/2020
RESPONSABLES		CRISTIAN ESPITIA			
DESCRIPCIÓN		El sistema debe tener un módulo con formularios web para registrar, actualizar, listar las escuelas deportivas.			

	<p>Registrar escuela: en un formulario se debe poder ingresar una nueva escuela deportiva, debe aparecer tanto la fecha en un capo visible como el nombre de usuario de quien está creando la escuela deportiva; los campos solicitados son:</p> <ul style="list-style-type: none"> ● Nombre ● Estado (activo e inactivo) <p>Tener un botón para guardar y que genere un mensaje de la creación correcta.</p> <p>Actualizar escuela: en un formulario se debe poder actualizar la escuela deportiva, debe aparecer tanto la fecha en un capo visible como el nombre de usuario de quien está actualizando la escuela deportiva; los campos solicitados son:</p> <ul style="list-style-type: none"> ● Escoger de una lista desplegable la escuela a actualizar ● Nombre de escuela ● Estado (activo e inactivo) <p>Tener un botón para actualizar y que genere un mensaje de la actualización correcta.</p> <p>Listado de escuelas: en un formulario se debe poder listar todas las escuelas deportivas tanto las activas como las inactivas. Los resultados deben mostrarse en una tabla ordenada mostrando nombre de la escuela y estado.</p>				
	<table border="1" style="width: 100%;"> <thead> <tr> <th style="width: 50%; text-align: center;">REQUERIMIENTO FUNCIONAL ASOCIADO</th> <th style="width: 50%; text-align: center;">REQUERIMIENTO NO FUNCIONAL ASOCIADO</th> </tr> </thead> <tbody> <tr> <td></td> <td> <ul style="list-style-type: none"> ▪ RNF002 ▪ RNF003 ▪ RNF004 ▪ RNF006 ▪ RNF009 </td> </tr> </tbody> </table>	REQUERIMIENTO FUNCIONAL ASOCIADO	REQUERIMIENTO NO FUNCIONAL ASOCIADO		<ul style="list-style-type: none"> ▪ RNF002 ▪ RNF003 ▪ RNF004 ▪ RNF006 ▪ RNF009
REQUERIMIENTO FUNCIONAL ASOCIADO	REQUERIMIENTO NO FUNCIONAL ASOCIADO				
	<ul style="list-style-type: none"> ▪ RNF002 ▪ RNF003 ▪ RNF004 ▪ RNF006 ▪ RNF009 				

Tabla 3- Requerimiento Funcional 003

DEFINICIÓN DEL REQUERIMIENTO		
NO. REQUERIMIENTO	RF004	El sistema debe gestionar los datos de los descriptores de logros y asociarlos adecuadamente a las escuelas registradas en el sistema.

PRIORIDAD	Alta	IMPORTANCIA	Alta	FECHA ELABORACIÓN	29/09/2020
RESPONSABLES	CRISTIAN ESPITIA				
DESCRIPCIÓN	<p>El sistema debe tener un módulo con formularios web para registrar, actualizar, listar los descriptores de logros de las escuelas deportivas.</p> <p>Registrar descriptor de logro: en un formulario se debe poder ingresar un nuevo descriptor de logro, debe aparecer tanto la fecha en un campo visible como el nombre de usuario de quien está creando el descriptor de logro. Los campos solicitados son:</p> <ul style="list-style-type: none"> ● Seleccionar de una lista desplegable la escuela que va a estar asociada ● Asignar el texto del descriptor ● Estado (activo e inactivo) <p>Tener un botón para guardar y que genere un mensaje de la creación correcta.</p> <p>Actualizar escuela: en un formulario se debe poder actualizar el descriptor de logro, debe aparecer tanto la fecha en un campo visible como el nombre de usuario de quien está actualizando la escuela deportiva; los campos solicitados son:</p> <ul style="list-style-type: none"> ● Seleccionar de una lista desplegable la escuela ● Seleccionar de la lista desplegable el descriptor a actualizar. ● Asignar el texto del descriptor ● Estado (activo e inactivo) <p>Tener un botón para actualizar y que genere un mensaje de la actualización correcta.</p> <p>Listado de descriptores: en un formulario se debe poder listar todos los descriptores de logro seleccionando la escuela deportiva desde una lista desplegable. Los resultados deben mostrarse en una tabla ordenada mostrando nombre de la escuela, el texto del descriptor y el estado.</p>				
	REQUERIMIENTO ASOCIADO	FUNCIONAL	REQUERIMIENTO ASOCIADO	NO FUNCIONAL ASOCIADO	
	✓ RF003		✓ RNF002 ✓ RNF003		

		✓ RNF004 ✓ RNF006 ✓ RNF009
--	--	----------------------------------

Tabla 4- Requerimiento Funcional 004

DEFINICIÓN DEL REQUERIMIENTO					
NO. REQUERIMIENTO		RF005	El sistema debe gestionar los datos básicos de los estudiantes para el posterior envío de informes.		
PRIORIDAD	Media	IMPORTANCIA	Media	FECHA ELABORACIÓN	29/09/2020
RESPONSABLES		CRISTIAN ESPITIA			
DESCRIPCIÓN		<p>El sistema debe tener un módulo con formularios web para registrar, actualizar, listar los datos básicos de los estudiantes</p> <p>Registrar nuevo estudiante: en un formulario se debe poder ingresar un nuevo estudiante, debe aparecer tanto la fecha en un campo visible como el nombre de usuario de quien está creando el descriptor de logro. Los campos solicitados son:</p> <ul style="list-style-type: none"> ● Nombres Completos ● Seleccionar la fecha de nacimiento ● Campo para digitar el correo de informe ● Seleccionar el curso del estudiante ● Código del estudiante ● Estado (activo inactivo) <p>Tener un botón para guardar y que genere un mensaje de la creación correcta.</p> <p>Actualizar estudiante: en un formulario se debe poder actualizar los datos básicos del estudiante, se debe poder buscar por nombre y arrojar una lista con las posibles coincidencias.</p> <p>En los resultados de la búsqueda debe mostrarse la opción para seleccionarlo, nombre del estudiante, curso, fecha de nacimiento y estado. Al momento de seleccionarlo debe abrir una ventana para actualizar los campos y mostrar la foto del estudiante:</p> <ul style="list-style-type: none"> ● Nombres Completos ● Seleccionar la fecha de nacimiento 			

	<ul style="list-style-type: none"> ● Campo para digitar el correo de informe ● Seleccionar el curso del estudiante ● Código del estudiante ● Estado (activo inactivo) <p>Tener un botón para guardar y que genere un mensaje de la creación correcta.</p> <p>Listado de estudiantes: en un formulario se debe poder listar todos los estudiantes seleccionando la escuela deportiva desde una lista desplegable. Los resultados deben mostrarse en una tabla ordenada mostrando nombre del estudiante, curso, fecha de nacimiento, Foto del estudiante, escuela en la que está inscrito.</p> <p>Listado de estudiantes no calificados: en un formulario se debe poder tener conocimiento de los estudiantes que no han sido calificados agrupados por escuela.</p>		
	REQUERIMIENTO ASOCIADO	FUNCIONAL	REQUERIMIENTO NO FUNCIONAL ASOCIADO
	✓ RF003		<ul style="list-style-type: none"> ✓ RNF002 ✓ RNF003 ✓ RNF004 ✓ RNF006 ✓ RNF008 ✓ RNF009

Tabla 5- Requerimiento Funcional 005

DEFINICIÓN DEL REQUERIMIENTO					
NO. REQUERIMIENTO	RF006		En el sistema se deben poder inscribir los estudiantes registrados en más de una escuela.		
PRIORIDAD	Baja	IMPORTANCIA	Media	FECHA ELABORACIÓN	29/09/2020
RESPONSABLES		CRISTIAN ESPITIA			
DESCRIPCIÓN		El sistema debe tener un módulo con formularios web para registrar un estudiante en varias escuelas deportivas seleccionando las que se requieran.			

<p>Se debe poder buscar el estudiante por nombre y seleccionarlo desde una lista ordenada que muestre el nombre, curso y fecha de nacimiento; luego en otra tabla escoger las escuelas a inscribir.</p> <p>Tener un botón para guardar y que genere un mensaje de la inscripción correcta.</p>	
REQUERIMIENTO FUNCIONAL ASOCIADO	REQUERIMIENTO NO FUNCIONAL ASOCIADO
<ul style="list-style-type: none"> ✓ RF003 ✓ RF005 	<ul style="list-style-type: none"> ✓ RNF003 ✓ RNF004 ✓ RNF005 ✓ RNF007

Tabla 6- Requerimiento Funcional 006

DEFINICIÓN DEL REQUERIMIENTO					
NO. REQUERIMIENTO		RF007	El sistema debe admitir la calificación de los estudiantes inscritos en una escuela y listar los que faltan por calificación.		
PRIORIDAD	Alta	IMPORTANCIA	Alta	FECHA ELABORACIÓN	29/09/2020
RESPONSABLES		CRISTIAN ESPITIA			
DESCRIPCIÓN		<p>El sistema debe tener un módulo con un formulario web donde se pueda escoger la escuela de formación y buscar los estudiantes inscritos en ella y que estén pendiente por calificación.</p> <p>Cuando se realice la búsqueda se deben listar los nombres de estudiantes en orden alfabético, curso, fecha de nacimiento, foto y escuela de formación.</p> <p>El sistema al momento de seleccionar al estudiante de la lista se debe abrir una nueva ventana debe traer los descriptores asociados a la escuela, con el fin de poder evaluar el estudiante bajo los criterios adecuados y distribuir en tres espacios verticales los datos:</p> <p>Primero: un título del informe con el texto “PRIMER SEMESTRE 2019-2020” luego, los datos del estudiante como el nombre, fecha de nacimiento, curso, escuela y foto.</p>			

<p>Segundo: Mostrar los descriptores de logro asociados a la escuela ordenándolos alfabéticamente y seguido de cada descriptor poner un campo de selección con los atributos En desarrollo (de fondo al texto colocar un color rojo), Competente (de fondo al texto colocar un color amarillo), Sobresaliente (de fondo al texto colocar un color verde).</p> <p>Tercero: Asignar dos espacios de texto uno para el aspecto positivo y el otro para el aspecto por mejorar del estudiante.</p> <p>Tener un botón para guardar la información validando que los campos de calificación de cada descriptor y aspectos positivos y por mejorar estén completamente diligenciados, luego debe generar un mensaje de la calificación correcta.</p>			
REQUERIMIENTO ASOCIADO	FUNCIONAL	REQUERIMIENTO NO FUNCIONAL ASOCIADO	
<ul style="list-style-type: none"> ✓ RF003 ✓ RF004 ✓ RF006 ✓ RF005 		<ul style="list-style-type: none"> ▪ RNF003 ▪ RNF004 ▪ RNF005 ▪ RNF007 ▪ RNF008 	

Tabla 7- Requerimiento Funcional 007

DEFINICIÓN DEL REQUERIMIENTO					
NO. REQUERIMIENTO		RF008	El sistema debe admitir la actualización de la calificación realizada de los estudiantes inscritos en una escuela		
PRIORIDAD	Alta	IMPORTANCIA	Alta	FECHA ELABORACIÓN	29/09/2020
RESPONSABLES		CRISTIAN ESPITIA			
DESCRIPCIÓN		<p>El sistema debe tener un módulo con un formulario web donde se pueda escoger la escuela de formación y buscar los estudiantes inscritos en ella y que tengan una calificación.</p> <p>Cuando se realice la búsqueda se deben listar los nombres de estudiantes en orden alfabético, curso, fecha de nacimiento, foto y escuela de formación.</p>			

<p>El sistema al momento de seleccionar al estudiante de la lista se debe abrir una nueva ventana debe traer los descriptores asociados a la escuela, con el fin de poder actualizar la evaluación del estudiante bajo los criterios adecuados y distribuir en tres espacios verticales los datos:</p> <p>Primero: un título del informe con el texto “PRIMER SEMESTRE 2019-2020” luego, los datos del estudiante como el nombre, fecha de nacimiento, curso, escuela y foto.</p> <p>Segundo: Mostrar los descriptores de logro asociados a la escuela ordenándolos alfabéticamente y seguido de cada descriptor poner un campo de selección con los atributos En desarrollo (de fondo al texto colocar un color rojo), Competente (de fondo al texto colocar un color amarillo), Sobresaliente (de fondo al texto colocar un color verde).</p> <p>Tercero: Asignar dos espacios de texto y traer los datos digitados, uno para el aspecto positivo y el otro para el aspecto por mejorar del estudiante.</p> <p>Tener un botón para actualizar la información validando que los campos de calificación de cada descriptor y aspectos positivos y por mejorar estén completamente diligenciados, luego debe generar un mensaje de la actualización correcta.</p>			
REQUERIMIENTO ASOCIADO	FUNCIONAL	REQUERIMIENTO NO FUNCIONAL ASOCIADO	
<ul style="list-style-type: none"> ✓ RF003 ✓ RF004 ✓ RF005 ✓ RF006 ✓ RF007 		<ul style="list-style-type: none"> ▪ RNF003 ▪ RNF004 ▪ RNF005 ▪ RNF007 ▪ RNF008 	

Tabla 8- Requerimiento Funcional 008

DEFINICIÓN DEL REQUERIMIENTO					
NO. REQUERIMIENTO		RF009	El sistema debe generar un reporte de los estudiantes con calificación y poder visualizar la respectiva calificación.		
PRIORIDAD	Media	IMPORTANCIA	Alta	FECHA ELABORACIÓN	29/09/2020

RESPONSABLES	CRISTIAN ESPITIA		
DESCRIPCIÓN	<p>En un listado mostrar los estudiantes con calificación realizada, se debe poder escoger un estudiante y visualizar la calificación, también mostrar en campos editables los aspectos positivos y por mejorar para poder realizar una corrección del texto ingresado, por parte un usuario con privilegios para que pueda ser revisada y actualizada.</p> <p>Listar los nombres de estudiantes en orden alfabético desde un campo de búsqueda que muestre los nombres de las escuelas y permita realizar la búsqueda de los estudiantes que se encuentren inscritos y con calificación realizada</p> <p>Mostrar en el listado: campo de selección, nombre del estudiante, curso, fecha de nacimiento, foto y escuela a la que se está realizando la calificación.</p> <p>Cuando se selecciona al estudiante de la lista se debe abrir una ventana con los logos de la escuela y distribuir en 4 espacios verticales los datos:</p> <p>Primero: un título del informe con el texto “PRIMER SEMESTRE 2019-2020” luego, los datos del estudiante como el nombre, fecha de nacimiento, curso, escuela y foto.</p> <p>Segundo: Mostrar los descriptores de logro asociados a la escuela ordenándolos alfabéticamente y seguido de cada descriptor Colocar en un campo los atributos: En desarrollo (de fondo al texto colocar un color rojo), Competente (de fondo al texto colocar un color amarillo), Sobresaliente (de fondo al texto colocar un color verde) según calificación correspondiente.</p> <p>Tercero: Mostrar en los dos espacios de texto los aspectos positivos y aspectos por mejorar para poder realizar una verificación del texto ingresado.</p> <p>Cuarto: mostrar el nombre del jefe de escuelas de formación y el nombre del usuario que realizo la calificación.</p>		
	REQUERIMIENTO ASOCIADO	FUNCIONAL	REQUERIMIENTO NO ASOCIADO
	<ul style="list-style-type: none"> ✓ RF003 ✓ RF004 		<ul style="list-style-type: none"> ▪ RNF003 ▪ RNF004

	<ul style="list-style-type: none"> ✓ RF005 ✓ RF006 ✓ RF007 	<ul style="list-style-type: none"> ▪ RNF005 ▪ RNF007 ▪ RNF008
--	---	--

Tabla 9- Requerimiento Funcional 009

DEFINICIÓN DEL REQUERIMIENTO					
NO. REQUERIMIENTO	RF010		Cuando se termine de actualizar los datos del estudiante escogido, transferir estos datos aun formato PDF para enviar por correo electrónico.		
PRIORIDAD	Alta	IMPORTANCIA	Alta	FECHA ELABORACIÓN	29/09/2020
RESPONSABLES		CRISTIAN ESPITIA			
DESCRIPCIÓN		<p>Cuando se finalice la actualización de los campos, se debe actualizar en la base de datos, posteriormente convertir esta información en un formato PDF guardarlo en una ubicación definida en el servidor, enviar vía correo electrónico al destinatario que tiene registrado el estudiante.</p> <p>El mensaje debe tener el archivo como adjunto, y con un mensaje personalizado; el cual será redactado por el personal en cargado por el colegio.</p> <p>Posterior al envío satisfactorio del correo buscar el archivo PDF generado y borrarlo para no aumentar el espacio en el servidor.</p>			
		REQUERIMIENTO ASOCIADO	FUNCIONAL	REQUERIMIENTO NO FUNCIONAL ASOCIADO	
		✓ RF009		<ul style="list-style-type: none"> ▪ RNF003 ▪ RNF004 ▪ RNF005 ▪ RNF007 ▪ RNF008 	

Tabla 10- Requerimiento Funcional 010

DEFINICIÓN DEL REQUERIMIENTO		
NO. REQUERIMIENTO	RF011	El sistema debe registrar el usuario que realice cambios o actualizaciones de la información registrada.

PRIORIDAD	Alta	IMPORTANCIA	Alta	FECHA ELABORACIÓN	29/10/2020
RESPONSABLES	CRISTIAN ESPITIA				
DESCRIPCIÓN	El sistema debe generar un registro en la base de datos para tener un control de los usuarios que realizan cambios en las escuelas deportivas, descriptores de logros, datos de estudiantes y datos de usuarios.				
	REQUERIMIENTO ASOCIADO	FUNCIONAL	REQUERIMIENTO ASOCIADO	NO FUNCIONAL ASOCIADO	
	✓ RF001		▪ RNF006		

Tabla 11- Requerimiento Funcional 011

Listado de requerimientos no funcionales

Son aquellos que no se refieren directamente a las funciones específicas que entrega el aplicativo, sino a las propiedades emergentes de este los cuales surgen de la necesidad del usuario.

A continuación, enumeramos los requerimientos que fueron obtenidos:

- RNF001: Las notas que se registran en el sistema no están sujetas al esquema curricular del colegio por lo que no hacen parte académica de la institución.
- RNF002: La aplicación debe desarrollarse en un lenguaje enfocado a la web.
- RNF003: El sistema debe tener una interfaz fácil de entender y operar.
- RNF004: Tener una conexión estable a internet.
- RNF005: El sistema debe disponer de una ayuda (manual de usuario) para guiar el usuario sobre el uso adecuado del sistema.
- RNF006: Garantizar al usuario el acceso a la información de acuerdo al perfil que posee.
- RNF007: el sistema tendrá un periodo de funcionamiento definido por las directivas del colegio para los usuarios con perfil profesor.
- RNF008: El sistema garantizará a los usuarios una seguridad en cuanto a la información diligenciada la cual será confidencial y de uso exclusivo del colegio para fines netamente informativos.
- RNF009: Validar que los campos no estén vacíos antes de guardar y mostrar una alerta al usuario para que diligencie el campo faltante.

Fase de diseño del sistema

Para nuestra aplicación web, es necesario abordar modelando los casos de uso, los cuales nos servirán para ilustrar las funcionalidades de nuestra aplicación. Estos modelos expresan el comportamiento del sistema afrontando un determinado requisito, se enfoca en los valores suministrados por los actores y el sistema.

Diagrama inicio de sesión

En la siguiente descripción grafica se muestran los actores que interactúan, determinando las operaciones que pueden realizar particularmente al iniciar sesión en la aplicación web.

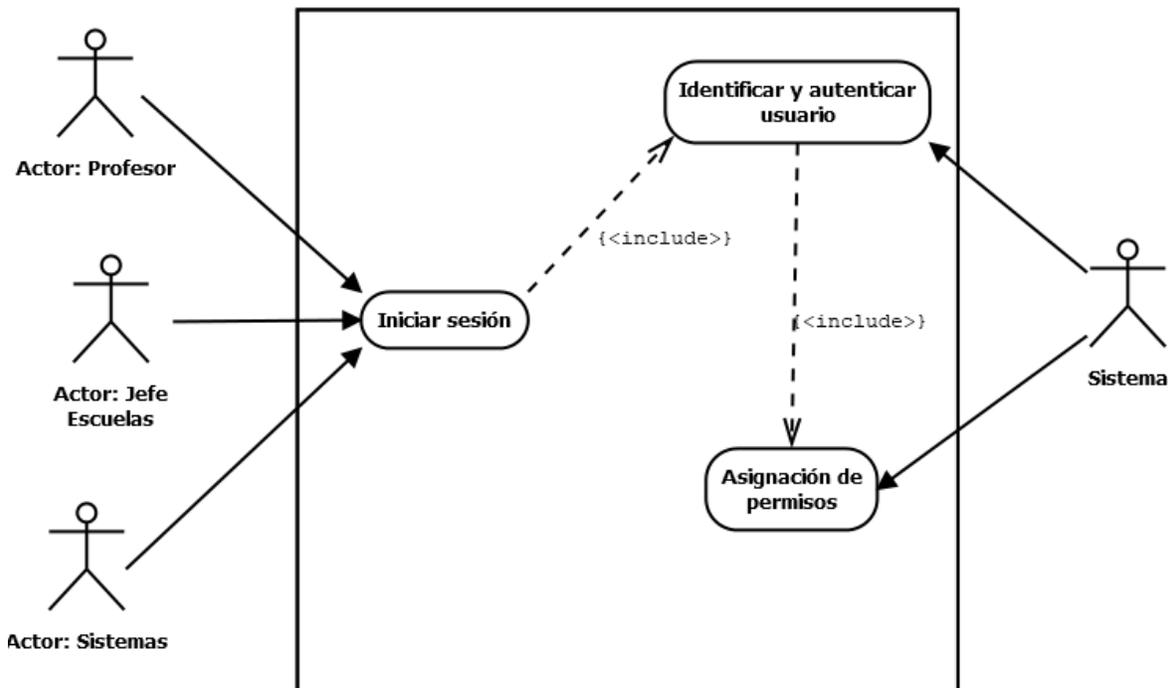


ILUSTRACIÓN 4- DIAGRAMA INICIO SESIÓN

FUENTE: PROPIA

Diagrama de usuarios

En el diagrama de usuarios se observan las principales funciones de los actores que interactúan con la gestión de los usuarios, los cuales pueden acceder a la información pertinente; particularmente el actor Jefe de escuelas puede acceder a registrar un usuario nuevo, actualizar los datos de los usuarios registrados, asignarles un perfil en la aplicación web y listar los usuarios registrados. Tanto el actor profesor y jefe de escuelas tienen autorización para actualizar su contraseña, todo esto se detalla en la siguiente ilustración.

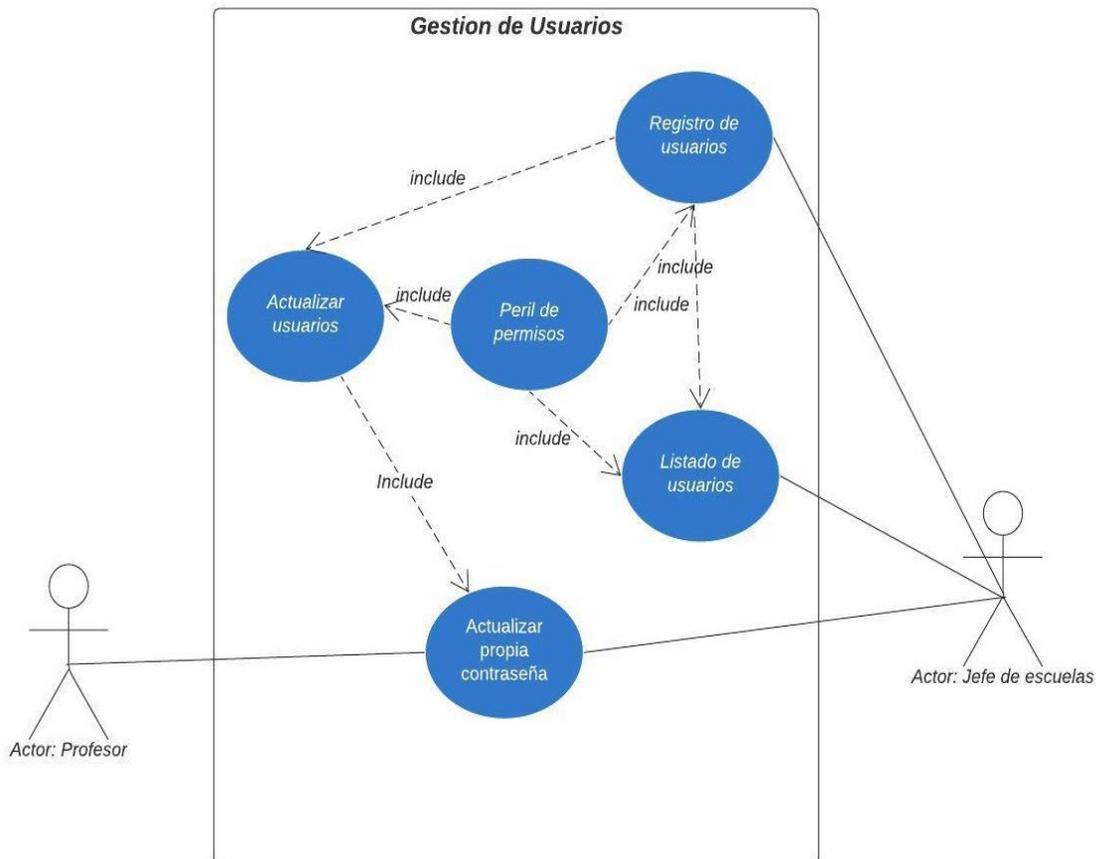


ILUSTRACIÓN 5- DIAGRAMA USUARIOS

FUENTE: PROPIA

Diagrama de estudiantes

En el diagrama de estudiantes se observan las principales funciones de los actores que interactúan, los cuales pueden acceder a la información pertinente dependiendo su rol en la aplicación; particularmente el actor Jefe de escuelas puede acceder a registrar estudiantes, actualizar los datos de los estudiantes, listar los estudiantes que tienen pendiente una calificación en la escuela inscrita. Tanto el actor profesor y jefe de escuelas tienen acceso a listar los estudiantes por escuela deportiva para tener una vista de los estudiantes, todo esto se detalla en la siguiente ilustración.

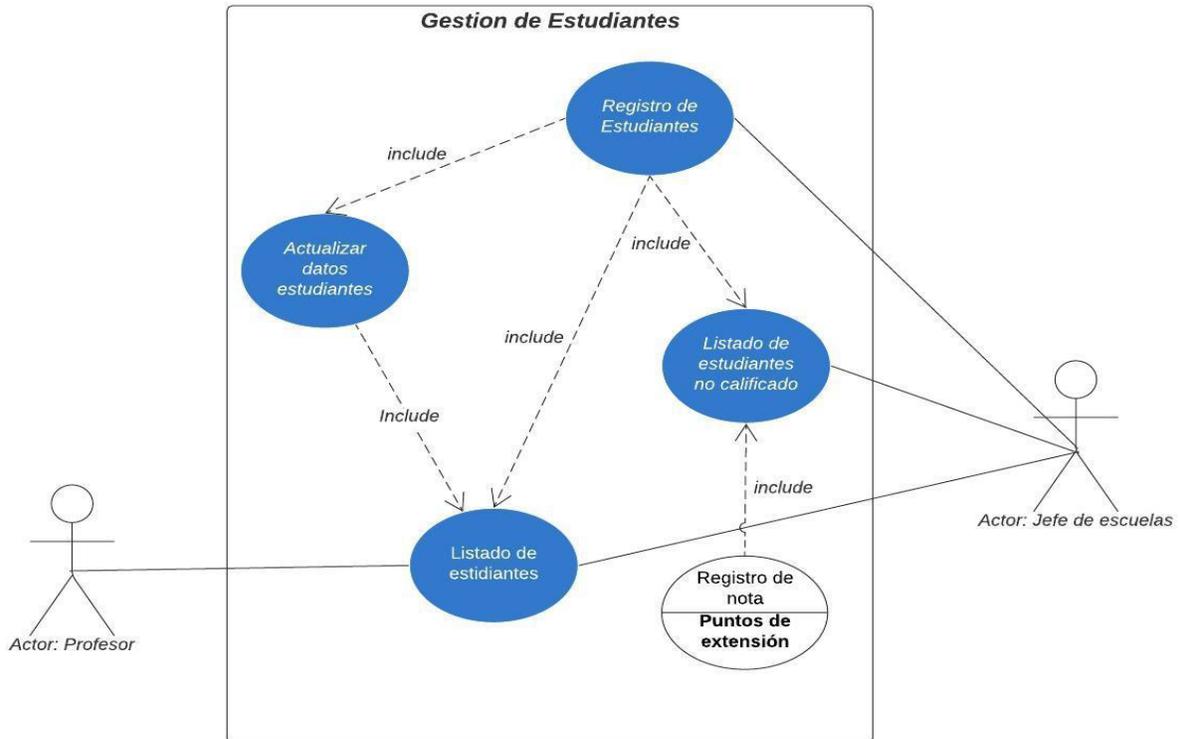


Ilustración 6- Diagrama de estudiantes

FUENTE: PROPIA

Diagrama escuelas deportivas

En el diagrama de escuelas deportivas se representan las principales funciones de los actores cuando interactúan con la gestión de las escuelas deportivas, los cuales obtienen a la información pertinente asociada a su perfil dentro de la aplicación; particularmente, los actores Jefe de escuelas y sistemas pueden acceder a registrar una escuela nueva, actualizarla. Los tres actores que interactúan con el modulo tienen autorización para consultar las escuelas que se encuentran registradas en la aplicación web, todo esto se detalla en la siguiente ilustración.

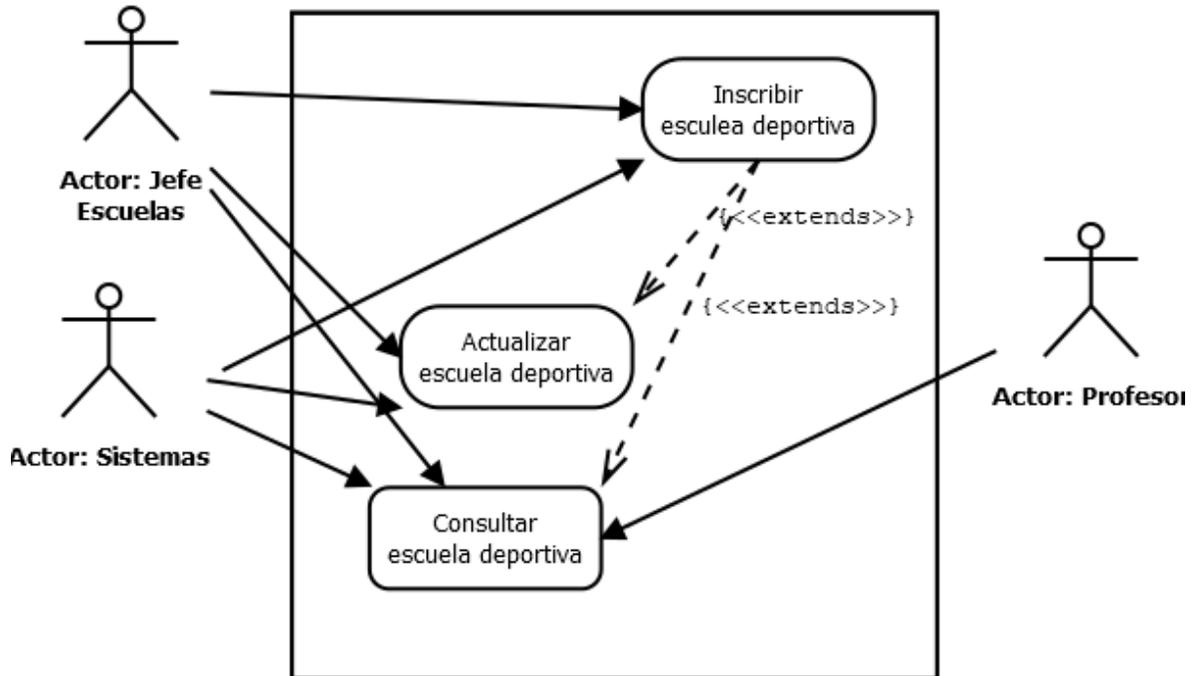


Ilustración 7- Diagrama escuelas deportivas
FUENTE: PROPIA

Diagrama descriptores de logro

En el diagrama de descriptores de logros se simbolizan las funciones de los actores al momento de interactuar con los descriptores de logros, los cuales obtienen a la información pertinente del actor sistema, la cual está asociada al perfil dentro de la aplicación; Los actores Jefe de escuelas y sistemas pueden acceder a registrar un nuevo descriptor asociándolo a una escuela deportiva donde el actor sistema debe validar que exista y este activa, actualizar el descriptor. Los tres actores que interactúan con el modulo tienen autorización para consultar los descriptores que se encuentran registradas en la aplicación web, todo esto se detalla en la siguiente ilustración.

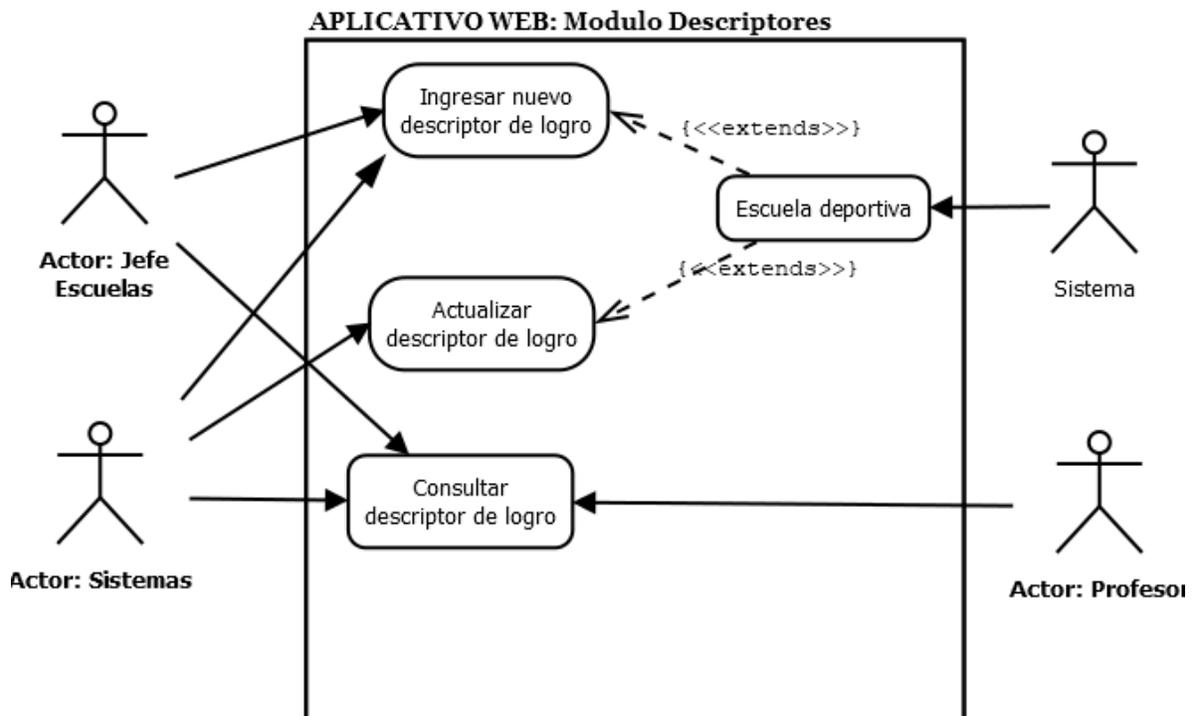


Ilustración 8- Diagrama Descriptores de Logro

FUENTE: PROPIA

Diagrama calificación

El diagrama de calificación representa gráficamente las funciones que tienen los actores profesor y jefe de escuelas puede realizar en la aplicación web; Un usuario con perfil del actor profesor puede visualizar un listado de los estudiantes inscritos en una escuela deportiva, de los cuales puede generar una calificación de los descriptores de logro asociados a la escuela deportiva y de la misma manera actualizarla, también posee acceso al listado de estudiantes no calificados. El actor jefe de escuelas tiene los mismos permisos que el actor profesor complementando la calificación realizada por el actor profesor generando el informe final y enviándolo a los correos asignados en los datos del estudiante, todo esto se detalla en la siguiente ilustración.

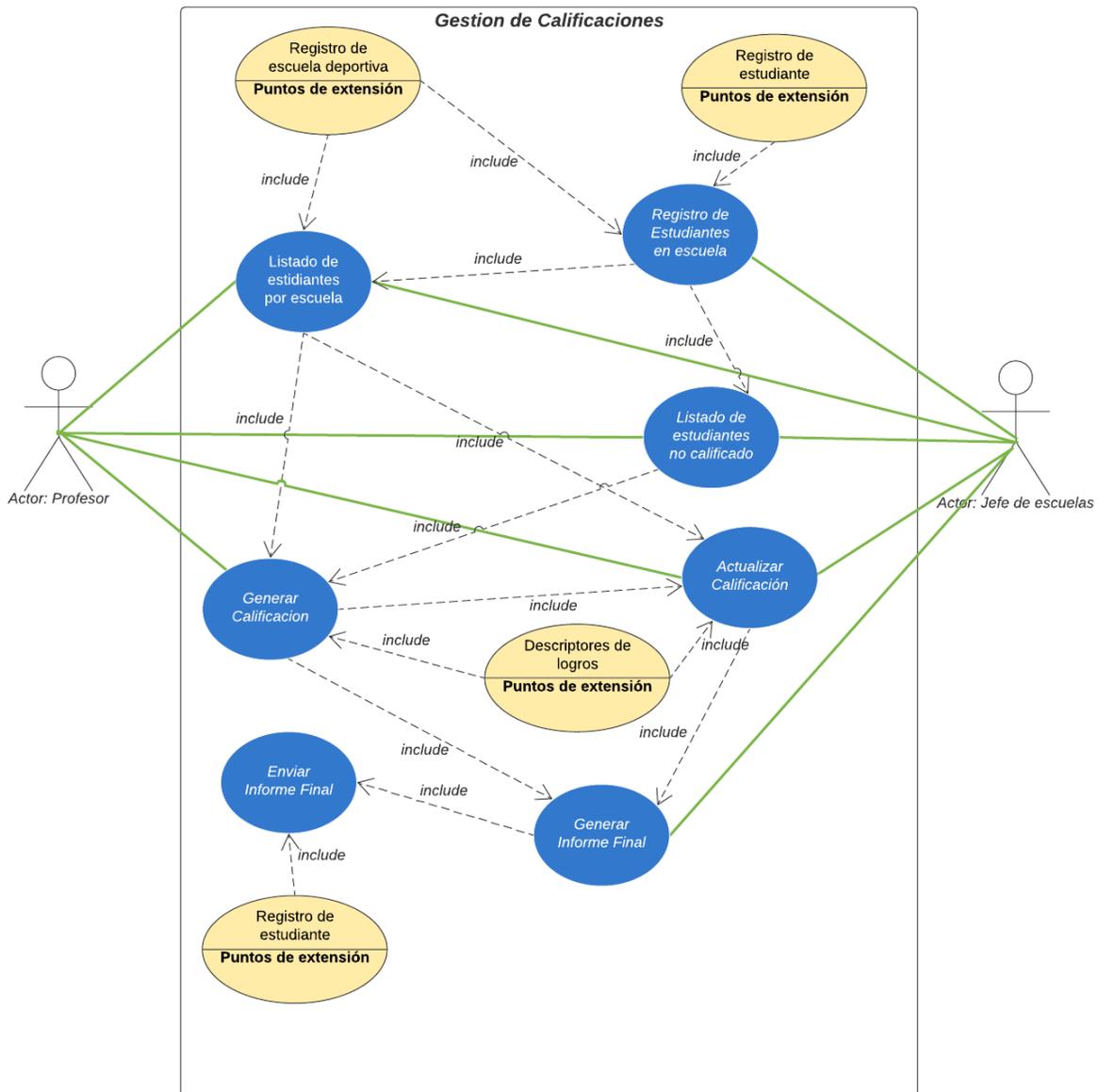


Ilustración 9- Diagrama Calificación

FUENTE: PROPIA

Diagrama físico

El siguiente diagrama describe los componentes de hardware y software que se desplegarán en el ambiente web. Describe los nodos de hardware utilizados, los protocolos de conexión entre nodos, los componentes de software que se utilizarán para la comunicación de toda la aplicación, la imagen de abajo muestra gráficamente el modelo abstracto con los nodos ejecutando los componentes necesarios para la aplicación web.

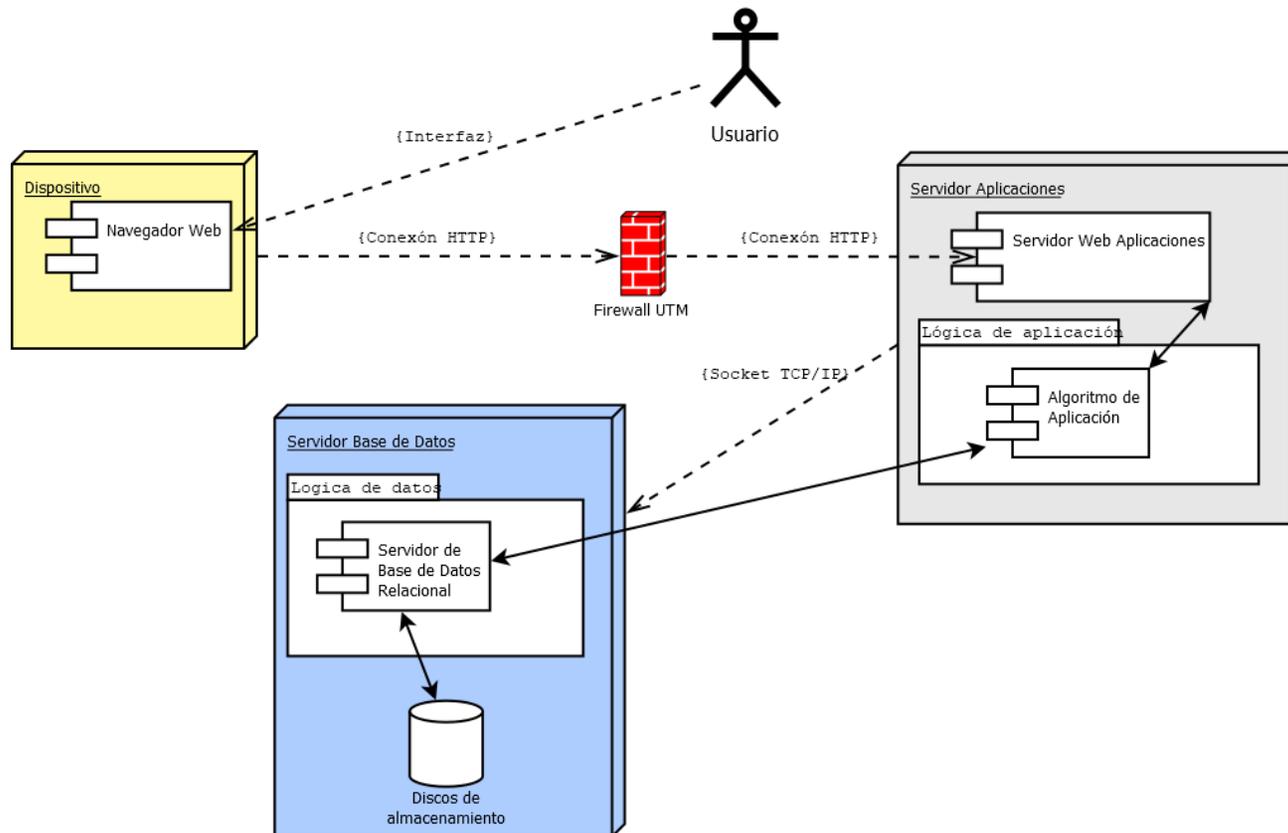


Ilustración 10- Diagrama físico
FUENTE: PROPIA

Diagrama de Clases

En el siguiente diagrama se presentan las clases de la aplicación web con sus objetos, métodos y relaciones dentro del sistema; los cuales intervienen logrando dibujar la aplicación a ser desarrollada, este esquema gráfico involucrado en la planeación y funcionamiento del sistema ayudará a modelar las relaciones implícitas que se deben tener en cuenta al momento de empezar la fase de codificación del software, todo lo anterior se define en la siguiente ilustración.

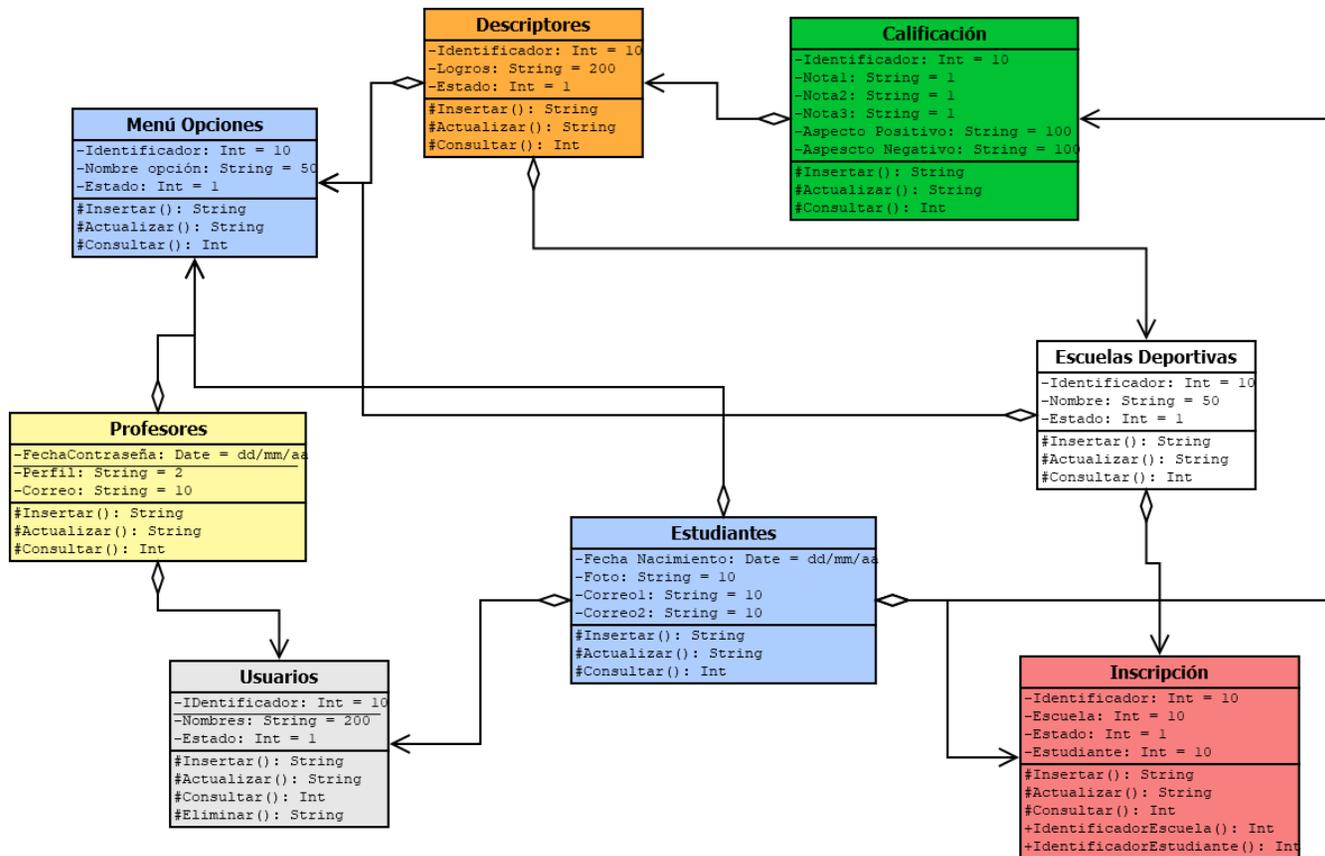


ILUSTRACIÓN 11- DIAGRAMA DE CLASES

FUENTE: PROPIA

Roles en la aplicación

Desde el levantamiento de los requisitos funcionales y no funcionales también se generó una tabla con los roles básicos que tendrán los usuarios registrados en la aplicación.

Rol en el sistema	Descripción
Jefe escuelas	Tiene acceso a todas las funciones de gestión de escuelas, descriptores, estudiantes Puede Actualizar calificaciones, enviar los informes Y además gestiona los usuarios del sistema.
Profesor(a)	Tiene la facultad de visualizar el listado de las escuelas, calificar los estudiantes y actualizar la calificación si se requiere. Puede visualizar el listado de los descriptores de logros y los estudiantes inscritos en las escuelas.
Sistemas	Tiene acceso a todas las herramientas, gestiona los usuarios del sistema.

Tabla 12- Roles de usuario

Modelado de datos

El modelado de datos es una manera de estructurar y organizar los datos para que se puedan utilizar fácilmente por las bases de datos. (inteldig.com, 2018)

Una vez completado el estudio de factibilidad se inició con el modelado de datos con el cual se generó el diccionario de datos inicial, se establecen las tablas bases de flujo de datos y que van a dar objetivo óptimo con el proyecto.

Modelo Relacional

En la ilustración 10, se representan los datos mediante tablas cuyas filas contienen los datos y las columnas la definición del campo de dato, conformado así una base de datos relacional; las tablas ya establecen una relación entre los diferentes datos que se rigen por las normas de normalización de datos evitando duplicidad de la información almacenada y garantiza la integridad referencial de los datos.

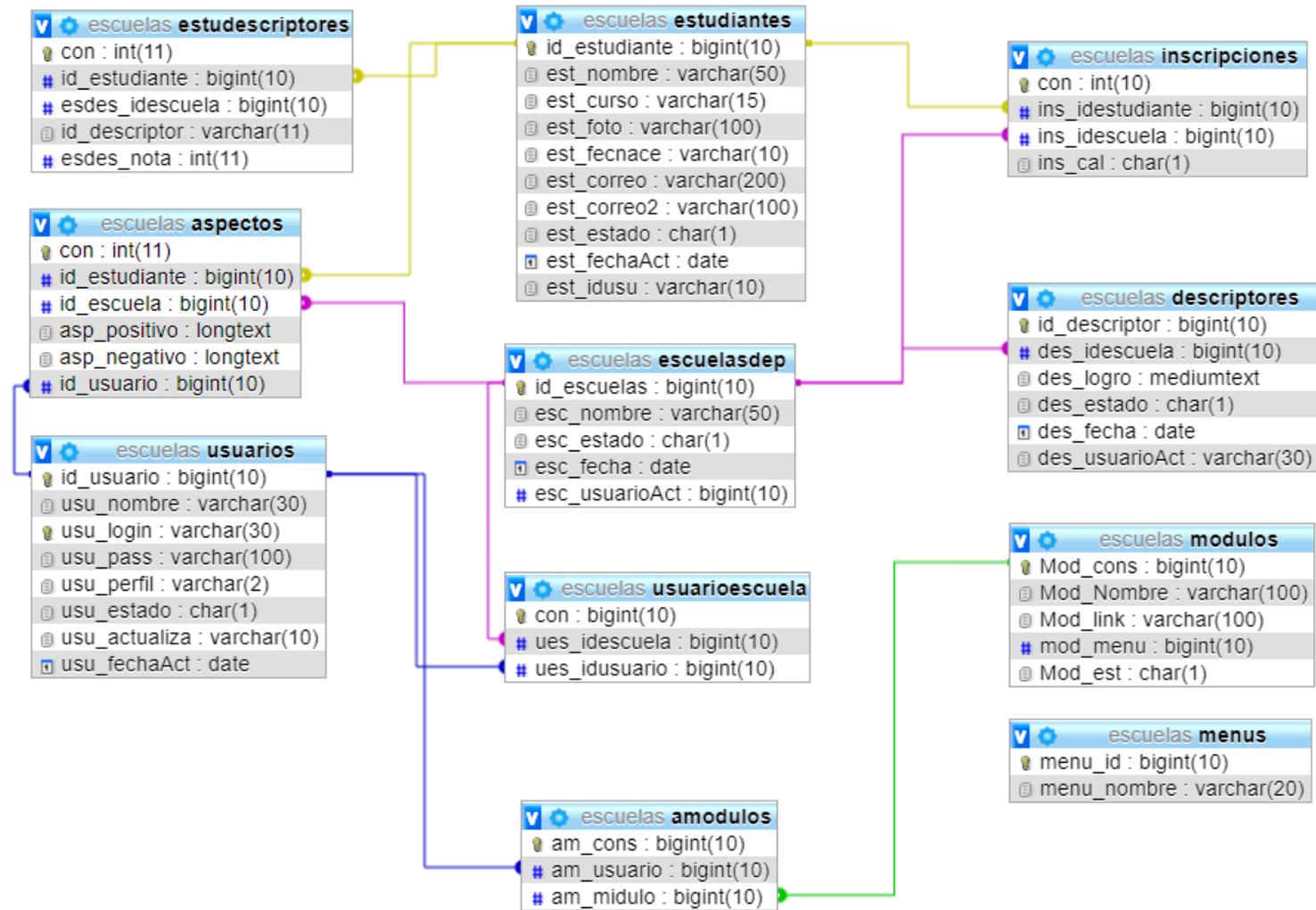


Ilustración 12- Modelo relacional
FUENTE: PROPIA

Diccionario de datos

A continuación, se describen todos los elementos que forman parte del flujo de datos en toda la aplicación de escuelas deportivas.

TABLA DE MÓDULOS				
Columna	Tipo	Nulo	Predeterminado	Enlaces a
am_cons (Primaria)	bigint(10)	No		
am_usuario	bigint(10)	No		usuarios -> id_usuario
am_midulo	bigint(10)	No		modulos -> Mod_cons

Tabla 13- Tabla ER: módulos

Índices de tabla modulos						
Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento
PRIMARY	BTREE	Sí	No	am_cons	37	A
am_usuario	BTREE	No	No	am_usuario	9	A
				am_midulo	37	A

Tabla 14- Índices de modulos

Tabla aspectos				
Columna	Tipo	Nulo	Predeterminado	Enlaces a
Con (Primaria)	int(11)	No		
id_estudiante	bigint(10)	No		estudiantes -> id_estudiante
id_escuela	bigint(10)	No		
asp_positivo	longtext	Sí	NULL	
asp_negativo	longtext	Sí	NULL	
id_usuario	bigint(10)	No		usuarios -> id_usuario

Tabla 15- Tabla ER: aspectos

Índices de tabla aspectos						
Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento
con	BTREE	Sí	No	con	0	A
id_estudiante	BTREE	No	No	id_estudiante	0	A
id_usuario	BTREE	No	No	id_usuario	0	A

Tabla 16- Índices de aspectos

Tabla descriptores				
Columna	Tipo	Nulo	Predeterminado	Enlaces a
id_descriptor (Primaria)	bigint(10)	No		
des_idescuela	bigint(10)	No		escuelasdep -> id_escuelas
des_logro	mediumtext	No		
des_estado	char(1)	No		
des_fecha	date	No		

des_usuarioAct	varchar(30)	Sí	NULL	
----------------	-------------	----	------	--

Tabla 17- Tabla ER: descriptores

Índices de tabla descriptores						
Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento
PRIMARY	BTREE	Sí	No	id_descriptor	225	A
des_idescuela	BTREE	No	No	des_idescuela	112	A

Tabla 18- Índices de descriptores

Tabla escuelaDep				
Columna	Tipo	Nulo	Predeterminado	Enlaces a
id_escuelas (<i>Primaria</i>)	bigint(10)	No		
esc_nombre	varchar(50)	No		
esc_estado	char(1)	No		
esc_fecha	date	No		
esc_usuarioAct	bigint(10)	Sí	NULL	

Tabla 19- Tabla ER: escuelaDep

Índices de tabla escuelaDep						
Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento
PRIMARY	BTREE	Sí	No	id_escuelas	58	A

Tabla 20- Índices de escuelaDep

Tabla estudescriptores				
Columna	Tipo	Nulo	Predeterminado	Enlaces a
con (<i>Primaria</i>)	int(11)	No		
id_estudiante	bigint(10)	No		estudiantes -> id_estudiante
esdes_idescuela	bigint(10)	No		
id_descriptor	varchar(11)	No		
esdes_nota	int(11)	No		

Tabla 21- Tabla ER: estudescriptores

Índices de tabla estudescriptores						
Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento
PRIMARY	BTREE	Sí	No	con	5	A
id_estudiante	BTREE	No	No	id_estudiante	2	A
				id_descriptor	5	A

Tabla 22- Índices de tabla estudescriptores

Tabla estudiantes				
Columna	Tipo	Nulo	Predeterminado	Enlaces a
id_estudiante (Primaria)	bigint(10)	No		
est_nombre	varchar(50)	No		
est_curso	varchar(15)	No		
est_foto	varchar(100)	No		
est_fecnace	varchar(10)	No		
est_correo	varchar(200)	Sí	NULL	
est_correo2	varchar(100)	Sí	NULL	
est_estado	char(1)	No	A	
est_fechaAct	date	Sí	NULL	
est_idusu	varchar(10)	Sí	NULL	

Tabla 23- Tabla ER: estudiantes

Índices de tabla estudiantes						
Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento
PRIMARY	BTREE	Sí	No	id_estudiante	486	A

Tabla 24- Índices tabla estudiantes

Tabla inscripciones				
Columna	Tipo	Nulo	Predeterminado	Enlaces a
con (Primaria)	int(10)	No		
ins_idestudiante	bigint(10)	No		estudiantes -> id_estudiante
ins_idescuela	bigint(10)	No		escuelasdep -> id_escuelas
ins_cal	char(1)	Sí	0	

Tabla 25- Tabla ER: inscripciones

Índices de tabla inscripciones						
Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento
PRIMARY	BTREE	Sí	No	con	594	A
ins_idestudiante	BTREE	No	No	ins_idestudiante	594	A
				ins_idescuela	594	A
ins_idescuela	BTREE	No	No	ins_idescuela	118	A

Tabla 26- Índices de tabla inscripciones

Tabla menus				
Columna	Tipo	Nulo	Predeterminado	Enlaces a
menu_id (Primaria)	bigint(10)	No		
menu_nombre	varchar(20)	No		

Tabla 27- Tabla ER: menus

Índices de tabla menus						
Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento
PRIMARY	BTREE	Sí	No	menu_id	4	A

Tabla 28- Índices de tabla menus

Tabla modulos				
Columna	Tipo	Nulo	Predeterminado	Enlaces a
Mod_cons (Primaria)	bigint(10)	No		
Mod_Nombre	varchar(100)	No		
Mod_link	varchar(100)	No		
mod_menu	bigint(10)	No		
Mod_est	char(1)	No	A	

Tabla 29- Tabla ER: modulos

Índices de tabla modulos						
Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento
PRIMARY	BTREE	Sí	No	Mod_cons	19	A
mod_menu	BTREE	No	No	mod_menu	9	A

Tabla 30- Índices de tabla modulos

Tabla usuarioescuela				
Columna	Tipo	Nulo	Predeterminado	Enlaces a
con (Primaria)	bigint(10)	No		
ues_idescuela	bigint(10)	No		escuelasdep -> id_escuelas
ues_idusuario	bigint(10)	No		usuarios -> id_usuario

Tabla 31- Tabla ER: usuarioescuela

Índices de tabla usuarioescuela						
Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento
PRIMARY	BTREE	Sí	No	con	4	A
ues_idescuelas	BTREE	No	No	ues_idescuela	4	A
				ues_idusuario	4	A
ues_idusuario	BTREE	No	No	ues_idusuario	4	A

Tabla 32- Índices de tabla usuarioescuela

Tabla usuarios				
Columna	Tipo	Nulo	Predeterminado	Enlaces a
id_usuario (Primaria)	bigint(10)	No		
usu_nombre	varchar(30)	No		
usu_login	varchar(30)	No		

usu_pass	varchar(30)	No		
usu_perfil	varchar(2)	No		
usu_estado	char(1)	No		
usu_actualiza	varchar(10)	Sí	NULL	
usu_fechaAct	date	Sí	NULL	

Tabla 33- Tabla ER: usuarios

Índices de tabla usuarios						
Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento
PRIMARY	BTREE	Sí	No	id_usuario	4	A
usu_login	BTREE	Sí	No	usu_login	4	A

Tabla 34- Índices de tabla usuarios

Modelo de Navegación

En este apartado la navegación del usuario está definida para que tenga una visión clara del sitio, con este, el usuario logrará desplazarse de acuerdo al rol asignado por el sistema encontrando los accesos correspondientes y que no desistan de usar la aplicación.

Perfil Jefe de escuelas y sistemas: En la ilustración 13, el modelo de navegación el usuario con este perfil podrá iniciar sesión e interactuar con todas las opciones del aplicativo web, una vez autenticado lo guiará al menú principal donde se enlazan a las diferentes interfaces:

- ✓ **Escuelas:** podrá ingresar, actualizar datos de las escuelas y generar listados; Calificar estudiantes inscritos y actualizar calificaciones realizadas.
- ✓ **Descriptores:** Podrá ingresar, actualizar y generar listados de los descriptores.
- ✓ **Estudiantes:** Podrá inscribir un nuevo estudiante, actualizar los datos de los estudiantes, listar los estudiantes inscritos, inscribir el estudiante en las diferentes escuelas, visualizar informe de estudiante por escuela, enviar el informe del estudiante, visualizar los estudiantes pendientes de calificación.
- ✓ **Administración:** gestionar usuarios de la plataforma, cambiar clave del usuario autenticado en la aplicación.

Perfil Instructor: En la ilustración 14, el modelo de navegación el usuario con este perfil podrá iniciar sesión e interactuar con todas las opciones del aplicativo web, una vez autenticado lo guiará al menú principal donde se enlazan a las diferentes interfaces:

- ✓ **Escuelas:** Generar listados; Calificar estudiantes inscritos y actualizar calificaciones realizadas.
- ✓ **Descriptores:** Podrá generar listados de los descriptores.
- ✓ **Estudiantes:** Podrá listar los estudiantes inscritos, visualizar informe de estudiante por escuela, visualizar los estudiantes pendientes de calificación.
- ✓ **Administración:** Cambiar clave del usuario autenticado en la aplicación.

Perfil sistemas: En la ilustración 15, el modelo de navegación el usuario con este perfil podrá iniciar sesión e interactuar con todas las opciones del aplicativo web, una vez autenticado lo guiará al menú principal donde se enlazan a las diferentes interfaces:

- ✓ **Escuelas:** podrá ingresar, actualizar datos de las escuelas y generar listados; Calificar estudiantes inscritos y actualizar calificaciones realizadas.
- ✓ **Descriptores:** Podrá ingresar, actualizar y generar listados de los descriptores.
- ✓ **Estudiantes:** Podrá inscribir un nuevo estudiante, actualizar los datos de los estudiantes, listar los estudiantes inscritos, visualizar informe de estudiante por escuela, visualizar los estudiantes pendientes de calificación.
- ✓ **Administración:** gestionar usuarios de la plataforma, cambiar clave del usuario autenticado en la aplicación.

MODELO DE NAVEGACIÓN PERFIL JEFE DE ESCUELAS

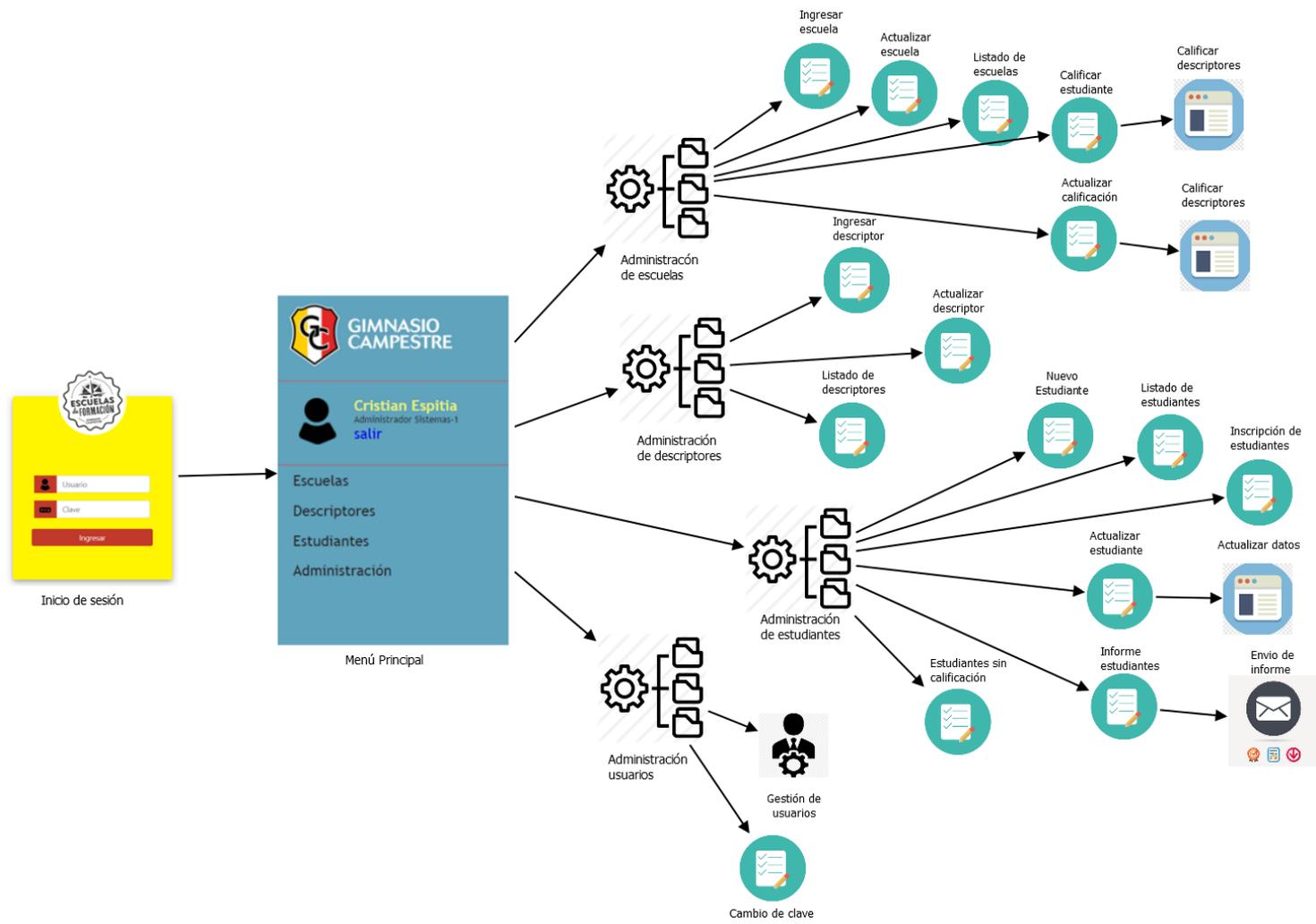


Ilustración 13- Modelo navegación perfil Jefe de escuelas

FUENTE: PROPIA

MODELO DE NAVEGACIÓN PERFIL INSTRUCTOR

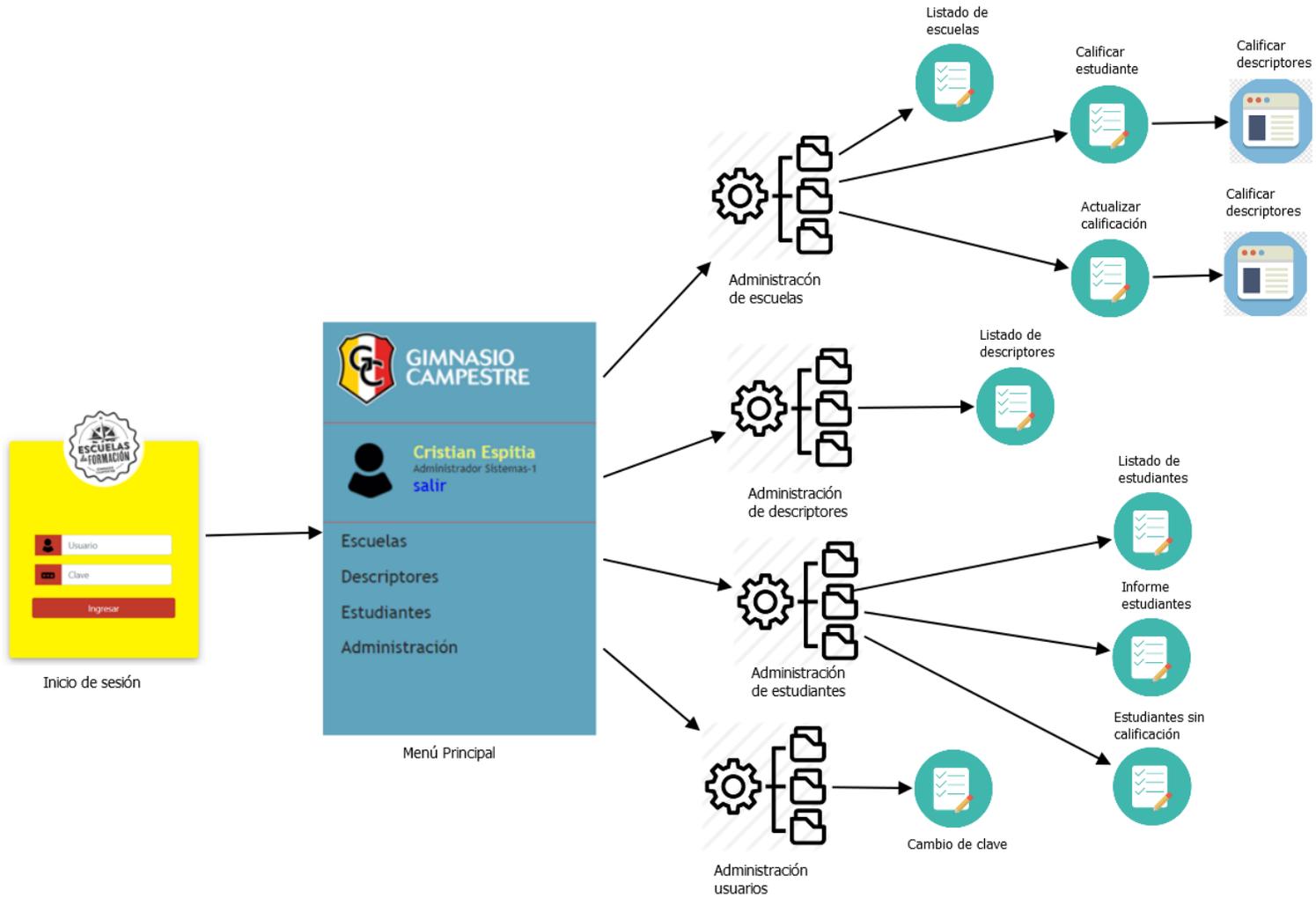


ILUSTRACIÓN 14- MODELO NAVEGACIÓN PERFIL INSTRUCTOR
FUENTE: PROPIA

MODELO DE NAVEGACIÓN PERFIL SISTEMAS

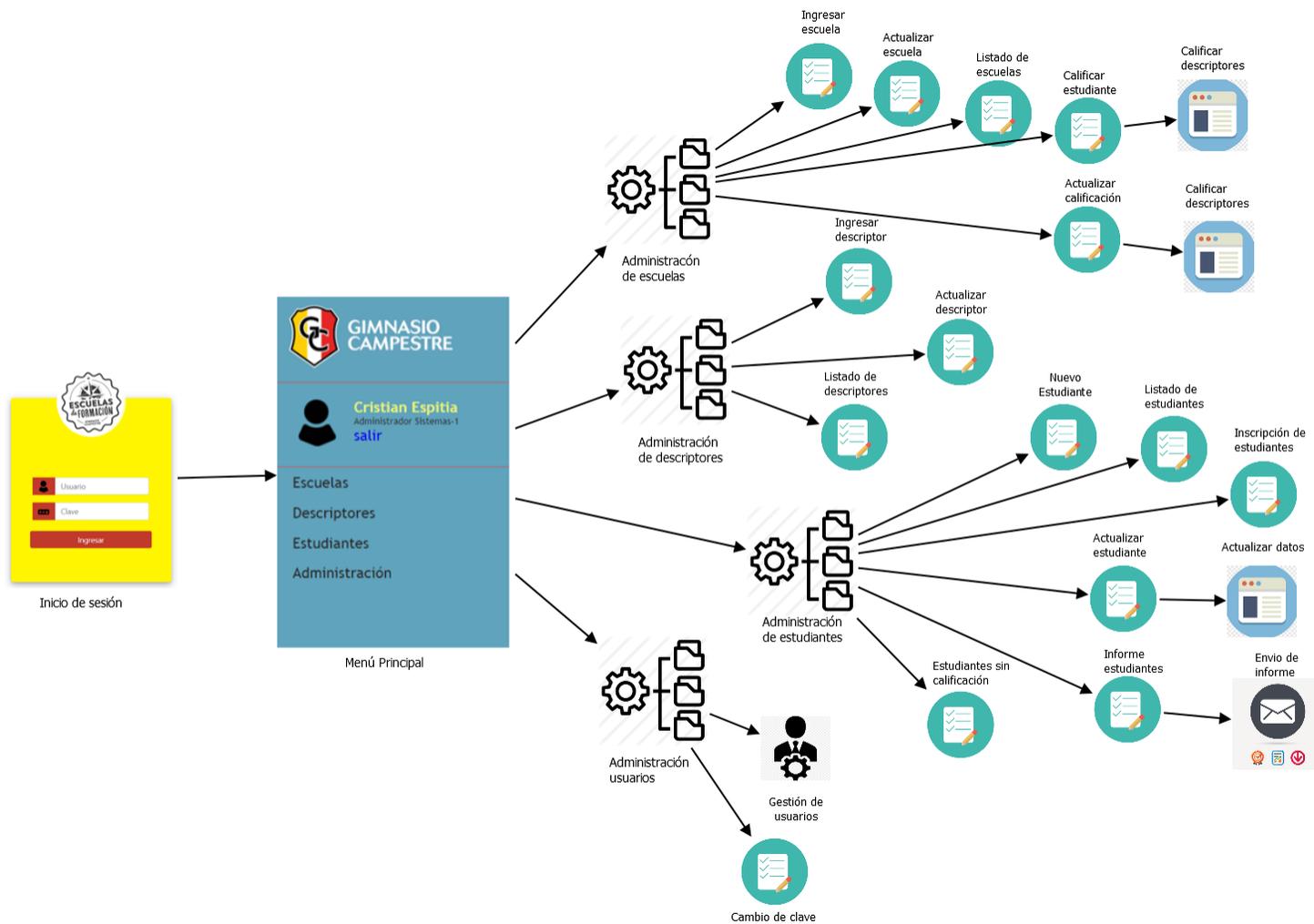


Ilustración 15- Modelo navegación perfil sistemas
FUENTE: PROPIA

Modelo de presentación

A continuación, se realiza un bosquejo de los estereotipos que son representados con gráficos realizando una descripción lo más simple posible, para la interacción del usuario con la aplicación dependiendo el rol asignado; con ello se expresa que dicho elemento gráfico está ubicado en la aplicación web.

INICIO DE SESIÓN

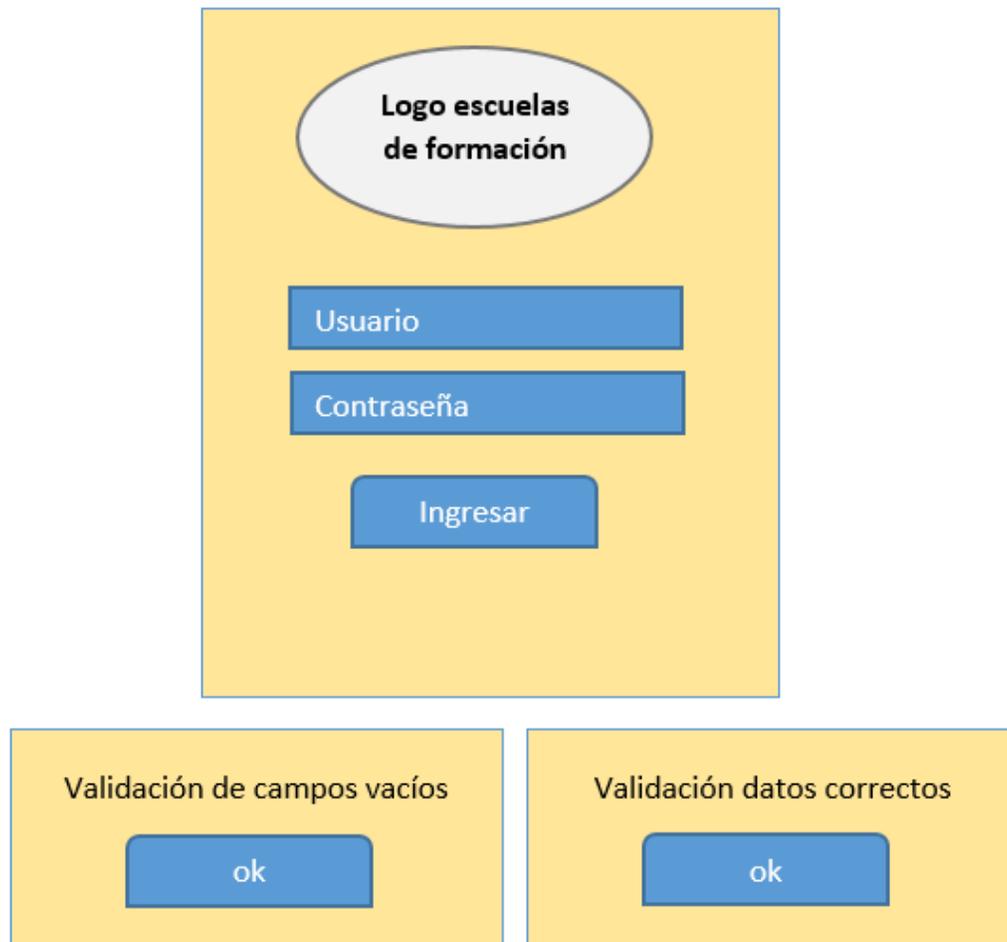


Ilustración 16- Modelo presentación: Inicio de sesión

FUENTE: PROPIA

OPCIONES APLICACIÓN

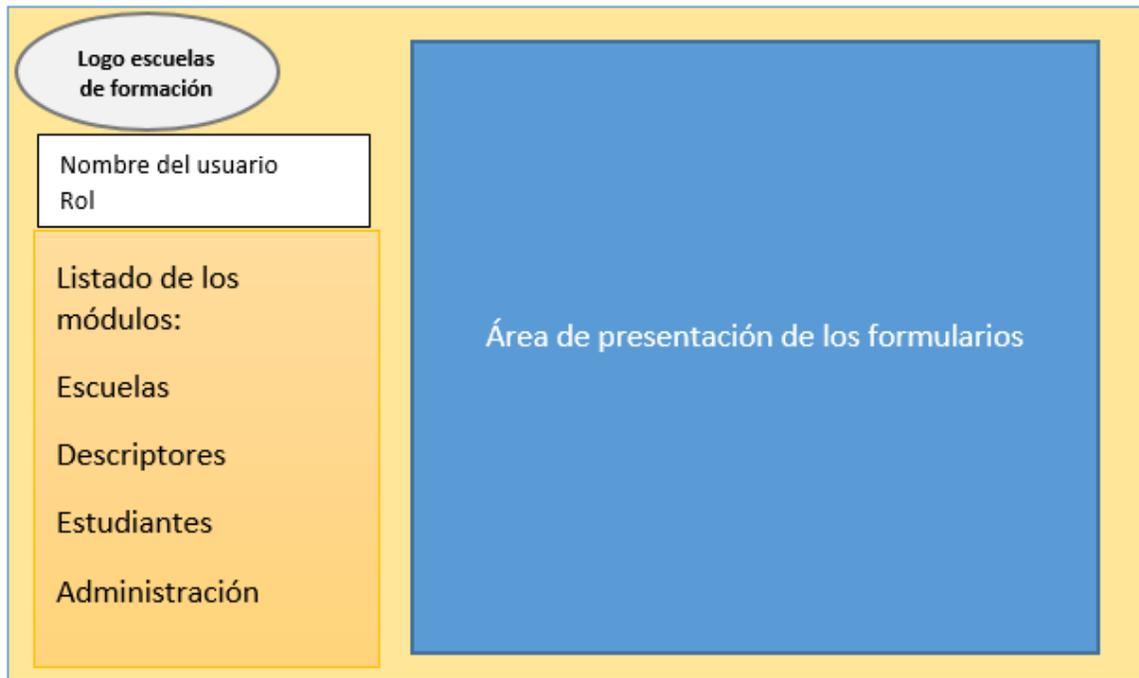


Ilustración 17- Modelo presentación opciones aplicación
FUENTE: PROPIA

MÓDULO ESCUELAS

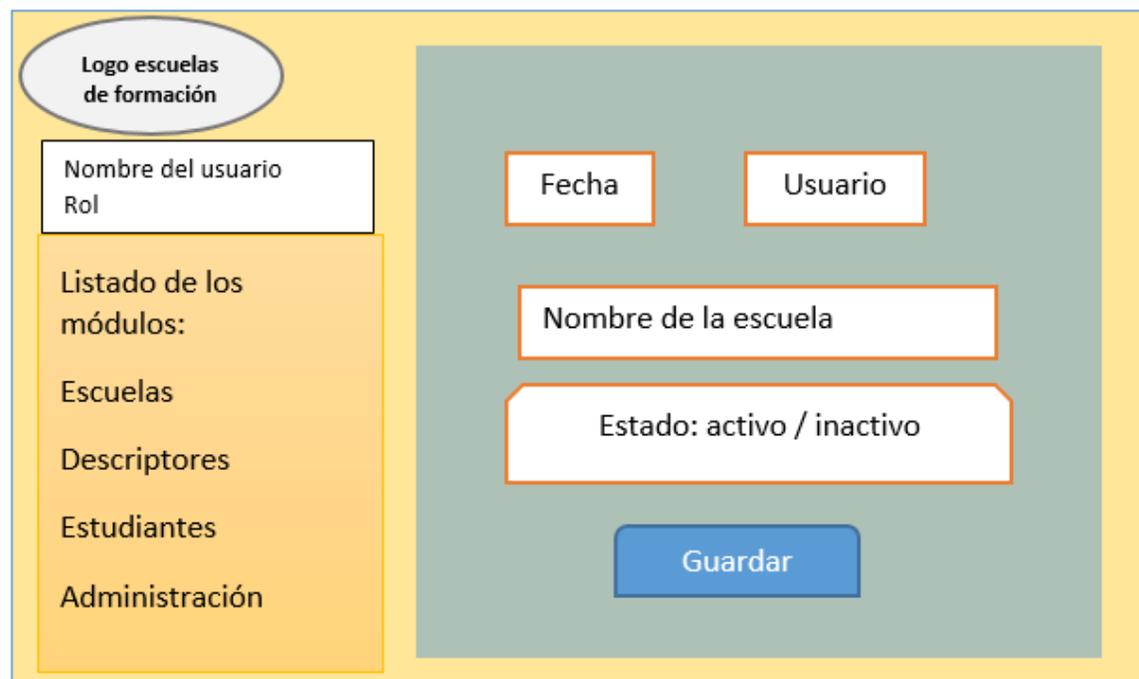


Ilustración 18- Modelo presentación: modulo escuelas
FUENTE: PROPIA

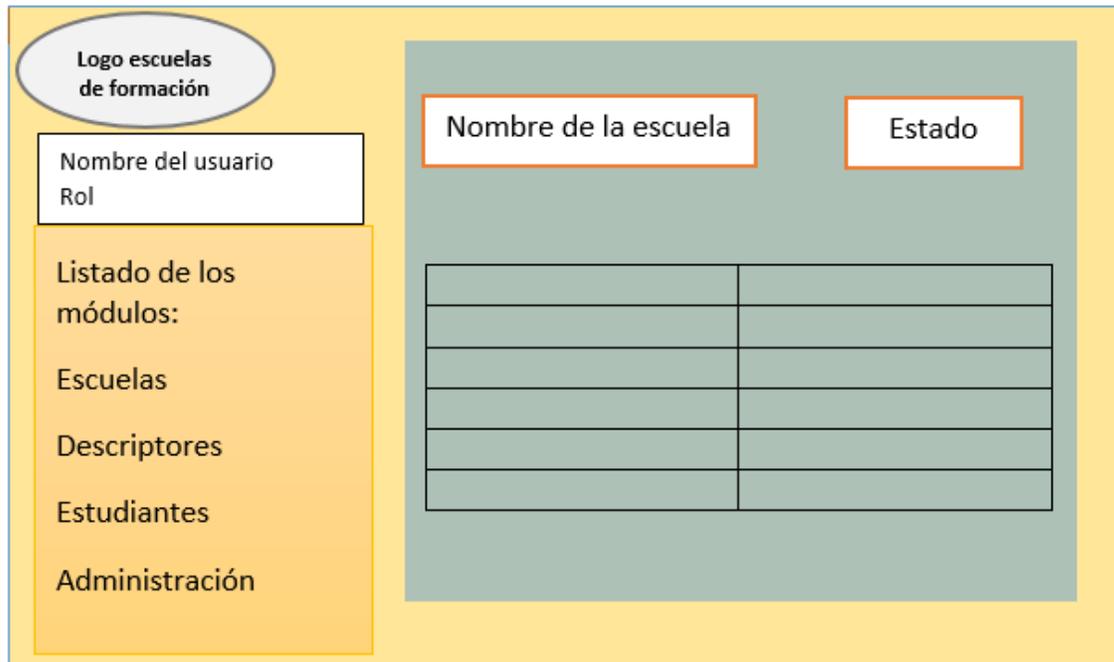


Ilustración 19- Modelo presentación: Listado escuelas

FUENTE: PROPIA

REGISTRAR DESCRIPTOR DE LOGRO

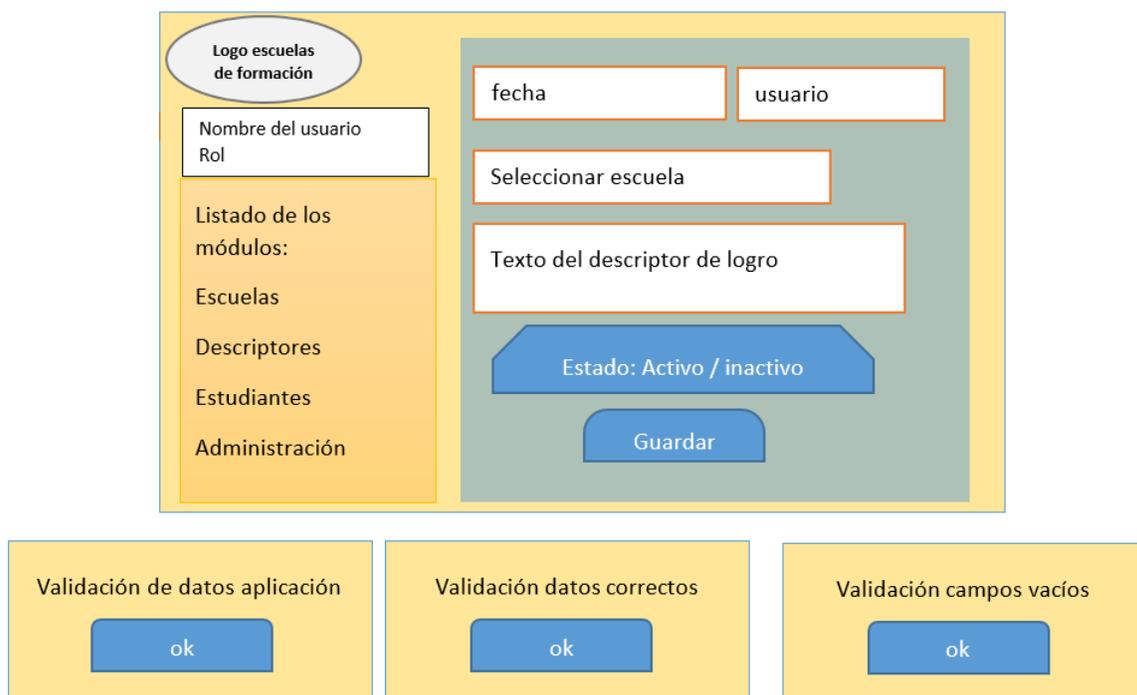


ILUSTRACIÓN 20- MODELO PRESENTACIÓN: REGISTRAR DESCRIPTOR DE LOGRO

FUENTE: PROPIA

CALIFICACIÓN DE ESTUDIANTE

Logo escuelas de formación

Nombre del usuario
Rol

Listado de los módulos:
Escuelas
Descriptorios
Estudiantes
Administración

Nombre de la escuela buscar

+	Nombre estudiante	Curso	Foto
+	Nombre estudiante	Curso	Foto
+	Nombre estudiante	Curso	Foto
+	Nombre estudiante	Curso	Foto
+	Nombre estudiante	Curso	Foto
+	Nombre estudiante	Curso	Foto

Ilustración 21- Modelo presentación: calificación de estudiante

FUENTE: PROPIA

Datos del estudiante:
Nombre
Fecha
Curso
Escuela

Foto del estudiante

Descriptor logro	Valoración		
	En Desarrollo	Competente	Sobresaliente
Logro 1			
Logro 2			
Logro 3			
Logro 4			

Aspectos positivos Aspectos por mejorar

Calificar

Validación de datos aplicación

ok

Validación datos correctos

ok

Validación campos vacíos

ok

Ilustración 22- Modelo presentación: calificación descriptorios

FUENTE: PROPIA

ENVIÓ DE INFORME POR CORREO ELECTRÓNICO

Datos del estudiante: Nombre Fecha Curso Escuela		Foto del estudiante	
Descriptor logro	Valoración		
	En Desarrollo	Competente	Sobresaliente
Logro 1			
Logro 2			
Logro 3			
Logro 4			
Aspectos positivos		Aspectos por mejorar	
Enviar informe			

Validación de correo enviado	Validación datos correctos
ok	ok

Ilustración 23- Modelo presentación: envío de informe
FUENTE: PROPIA

INSCRIPCIÓN DE ESTUDIANTE

Logo escuelas de formación

Nombre del usuario
Rol

Listado de los módulos:
Escuelas
Descriptores
Estudiantes
Administración

Nombre estudiante buscar

	Nombre estudiante	Curso	Foto	Nombre escuela	inscribir
+	Nombre estudiante	Curso	Foto	Escuela A	ok

Validación de datos aplicación Validación datos correctos Validación campos vacíos

ok ok ok

Ilustración 24- Modelo presentación: inscripción de estudiante
FUENTE: PROPIA

Fase de codificación del software

Para llevar a cabo esta aplicación web y que responda a las necesidades de este problema se utilizarán distintos lenguajes de programación web, el software necesario para originar el código de la aplicación y las bases de datos para almacenar todos los datos estará a disposición del Colegio para su uso.

Un lenguaje de programación es un lenguaje formal que especifica una serie de instrucciones (comando – códigos) para que una computadora produzca y procese diversas clases de datos. Los lenguajes de programación pueden usarse para crear programas que lleven a la práctica algoritmos específicos los cuales controlan el comportamiento físico y lógico de una computadora.

Está formado por un conjunto de símbolos y reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos y expresiones. Al proceso por el cual se escribe, se prueba, se depura, se compila (de ser necesario) y se mantiene el código fuente de un programa informático se le llama programación. (Wikipedia, 2019)

Para este proyecto se escogió como lenguaje de desarrollo PHP (acrónimo recursivo de *PHP: Hypertext Preprocessor*) es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML. Por lo que facilita una referencia de aprendizaje baja porque es un lenguaje totalmente libre y

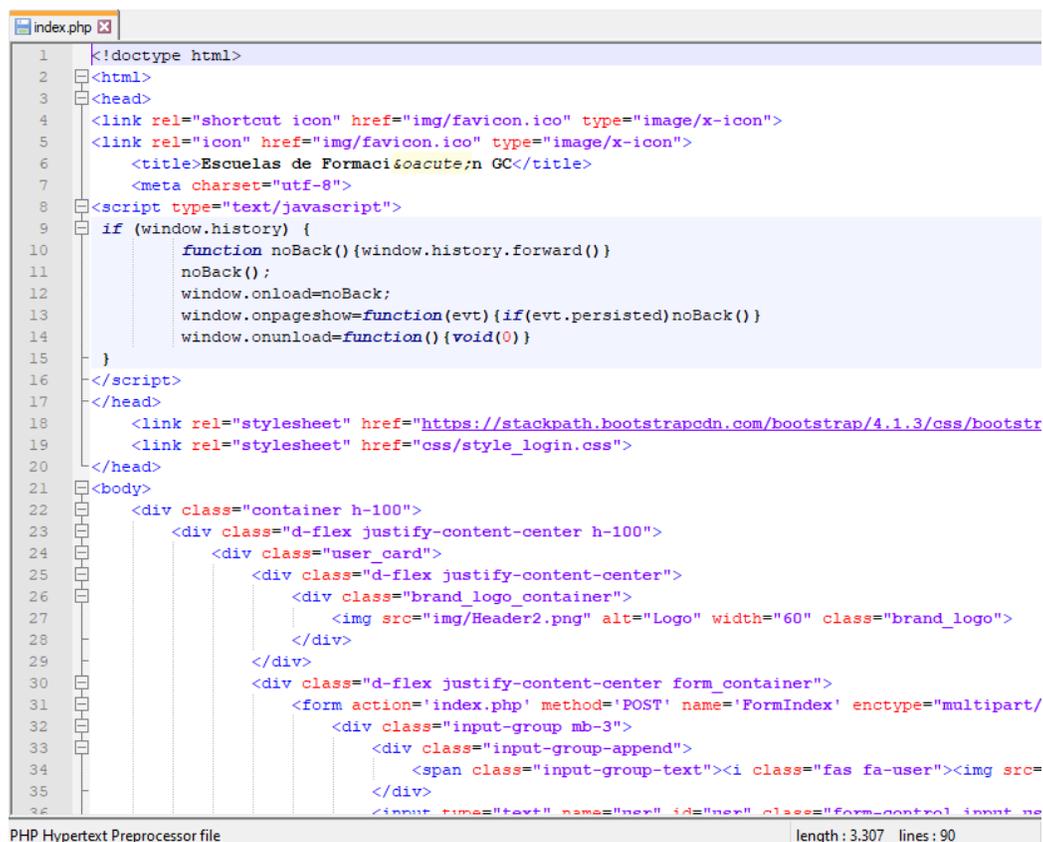
abierto, es de fácil acceso a distintos motores de bases de datos, los entornos de desarrollo son de rápida y completa configuración adicionalmente la documentación es amplia y con comunidad de ayudas y referencias. (Peter Cowburn, 2019)

HTML

HTML (Lenguaje de Marcas de Hipertexto, del inglés *HyperText Markup Language*) es el componente más básico de la Web. Define el significado y la estructura del contenido web. Además de HTML, generalmente se utilizan otras tecnologías para describir la apariencia/presentación de una página web (CSS) o la funcionalidad/comportamiento (JavaScript) (Developer Mozilla org., 2020); Este lenguaje permite introducir texto, imágenes u otros elementos en el diseño que sedeamos plasmar.

Gracias a este lenguaje, la aplicación web solo contiene texto de forma que, si incluimos imágenes, vídeos u otros elementos, estos son traídos al navegador, siendo el navegador web instalado en los dispositivos el encargado de interpretar el código y ajustar todos los elementos que finalmente visualiza el usuario. Además, este permite la inclusión de scripts, que pueden ser aplicados con JavaScript y que amplían aún más sus posibilidades.

En el siguiente fragmento de código se muestra la distribución de etiquetas HTML que son utilizadas en la página de inicio una vez han ingresado en la aplicación Web:



```
1 <!doctype html>
2 <html>
3 <head>
4 <link rel="shortcut icon" href="img/favicon.ico" type="image/x-icon">
5 <link rel="icon" href="img/favicon.ico" type="image/x-icon">
6 <title>Escuelas de Formación GC</title>
7 <meta charset="utf-8">
8 <script type="text/javascript">
9 if (window.history) {
10 function noBack() {window.history.forward()}
11 noBack();
12 window.onload=noBack;
13 window.onpageshow=function(evt) {if(evt.persisted)noBack()}
14 window.onunload=function() {void(0)}
15 }
16 </script>
17 </head>
18 <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/css/bootstr
19 <link rel="stylesheet" href="css/style_login.css">
20 </head>
21 <body>
22 <div class="container h-100">
23 <div class="d-flex justify-content-center h-100">
24 <div class="user_card">
25 <div class="d-flex justify-content-center">
26 <div class="brand_logo_container">
27 
28 </div>
29 </div>
30 <div class="d-flex justify-content-center form_container">
31 <form action='index.php' method='POST' name='FormIndex' enctype='multipart/
32 <div class="input-group mb-3">
33 <div class="input-group-append">
34 <span class="input-group-text"><i class="fas fa-user"><img src=
35 </div>
36 <input type="text" name="next" id="next" class="form-control input ne
```

Ilustración 25- Fragmento de código HTML

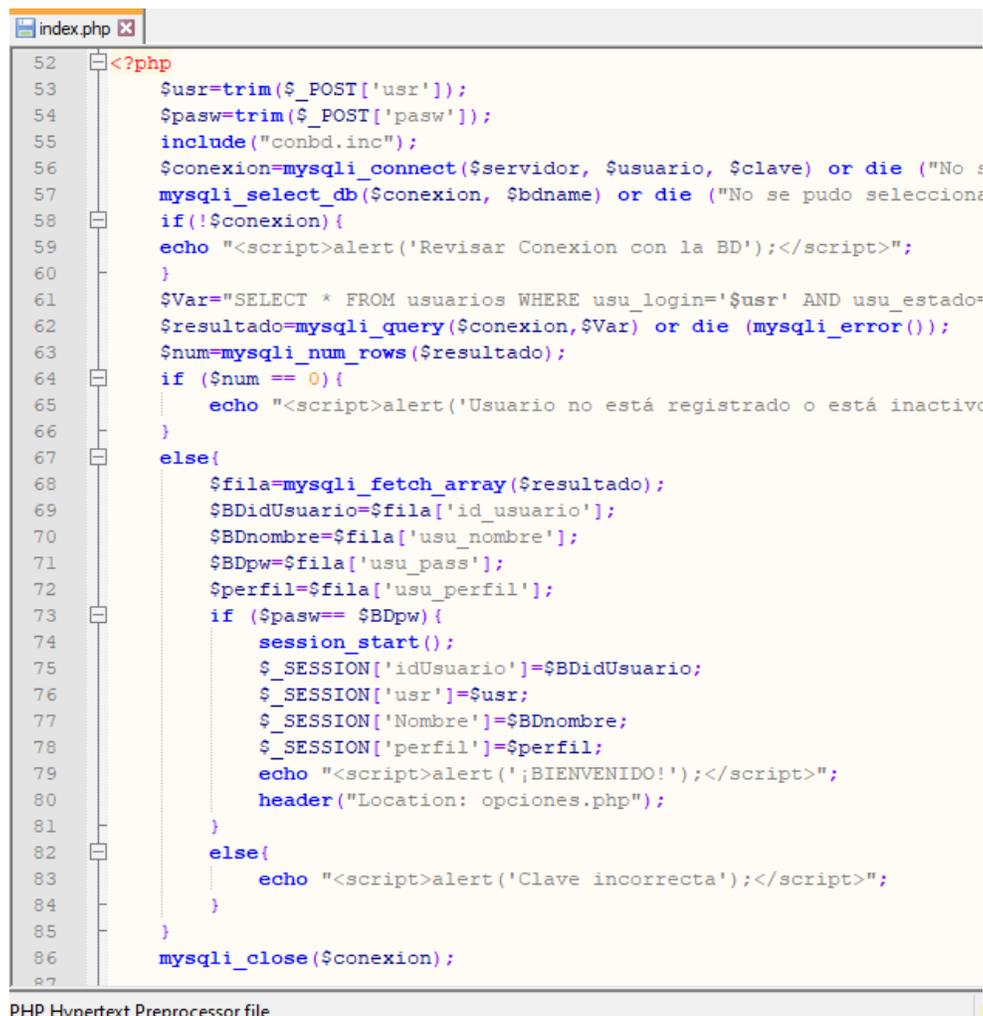
FUENTE: PROPIA

PHP

PHP (acrónimo recursivo de PHP: Hypertext Preprocessor) es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML. (Php.net, 2021)

Este es el lenguaje elegido del lado del servidor para interpretar, analizar y generar las partes dinámicas de la aplicación momento antes de que se envíe al navegador al usuario. Estas se ejecutan en el servidor y pueden realizar accesos a bases de datos, conexiones en red, y otras tareas para crear la página final que verá el usuario. Ya que se trata de código abierto y fácilmente de encontrar documentación, información de expertos y tutoriales; por ser software libre no hay que incurrir en compras de software específico para el desarrollo de esta aplicación web.

En el siguiente fragmento de código se muestra la conexión a la base de datos y la parametrización de opciones que tiene el usuario autenticado:



```
index.php x
52 <?php
53 $usr=trim($_POST['usr']);
54 $pasw=trim($_POST['pasw']);
55 include("conbd.inc");
56 $conexion=mysqli_connect($servidor, $usuario, $clave) or die ("No se pudo conectar");
57 mysqli_select_db($conexion, $bdname) or die ("No se pudo seleccionar base de datos");
58 if(!$conexion){
59     echo "<script>alert('Revisar Conexion con la BD');</script>";
60 }
61 $Var="SELECT * FROM usuarios WHERE usu_login='$usr' AND usu_estado='activo'";
62 $resultado=mysqli_query($conexion,$Var) or die (mysqli_error());
63 $num=mysqli_num_rows($resultado);
64 if ($num == 0){
65     echo "<script>alert('Usuario no está registrado o está inactivo');</script>";
66 }
67 else{
68     $fila=mysqli_fetch_array($resultado);
69     $BDIdUsuario=$fila['id_usuario'];
70     $BDNombre=$fila['usu_nombre'];
71     $BDpw=$fila['usu_pass'];
72     $perfil=$fila['usu_perfil'];
73     if ($pasw== $BDpw){
74         session_start();
75         $_SESSION['idUsuario']=$BDIdUsuario;
76         $_SESSION['usr']=$usr;
77         $_SESSION['Nombre']=$BDNombre;
78         $_SESSION['perfil']=$perfil;
79         echo "<script>alert(';BIENVENIDO!');</script>";
80         header("Location: opciones.php");
81     }
82     else{
83         echo "<script>alert('Clave incorrecta');</script>";
84     }
85 }
86 mysqli_close($conexion);
87
```

PHP Hypertext Preprocessor file

Ilustración 26- Fragmento de código PHP

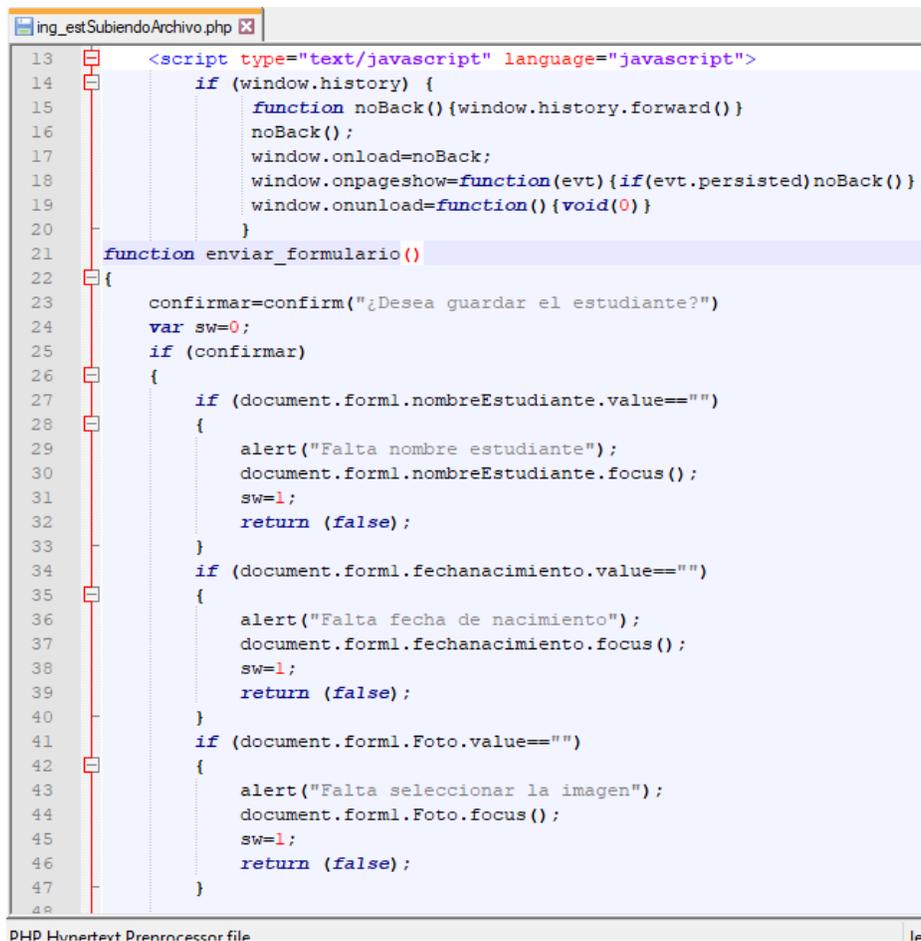
FUENTE: PROPIA

JAVASCRIPT

JavaScript es un lenguaje de programación o de secuencias de comandos que te permite implementar funciones complejas en páginas web, cada vez que una página web hace algo más que sentarse allí y mostrar información estática para que la veas, muestra oportunas actualizaciones de contenido, mapas interactivos, animación de Gráficos 2D/3D, desplazamiento de máquinas reproductoras de vídeo, etc., puedes apostar que probablemente JavaScript está involucrado. Es la tercera capa del pastel de las tecnologías web estándar, dos de las cuales (HTML y CSS) hemos cubierto con mucho más detalle en otras partes del Área de aprendizaje. (Developer Mozilla Org, 2020)

Este lenguaje es elegido del lado del usuario, ya que es el navegador el que soporta toda la carga y estará ejecutándose al momento de realizar la carga de información. Gracias a su amplia compatibilidad resulta muy útil generando una capa inicial para realizar validaciones de datos.

En el siguiente fragmento de código se valida por JavaScript los parámetros cuando se envía un formulario.

The image shows a screenshot of a code editor window titled 'ing_estSubiendoArchivo.php'. The code is JavaScript and is used for form validation. It starts with a script tag and includes a 'noBack()' function to prevent the browser from going back. The main function, 'enviar_formulario()', contains several conditional checks: 1) A confirmation dialog asking '¿Desea guardar el estudiante?'. 2) A check for an empty 'nombreEstudiante' field, which triggers an alert and focuses the field. 3) A check for an empty 'fechanacimiento' field, which triggers an alert and focuses the field. 4) A check for an empty 'Foto' field, which triggers an alert and focuses the field. The code uses 'document.form1' to access form elements and 'alert()' for user notifications. Line numbers 13 through 48 are visible on the left side of the editor.

```
13 <script type="text/javascript" language="javascript">
14     if (window.history) {
15         function noBack() {window.history.forward()}
16         noBack();
17         window.onload=noBack;
18         window.onpageshow=function(evt) {if(evt.persisted)noBack()}
19         window.onunload=function() {void(0)}
20     }
21     function enviar_formulario()
22     {
23         confirmar=confirm("¿Desea guardar el estudiante?")
24         var sw=0;
25         if (confirmar)
26         {
27             if (document.form1.nombreEstudiante.value=="")
28             {
29                 alert("Falta nombre estudiante");
30                 document.form1.nombreEstudiante.focus();
31                 sw=1;
32                 return (false);
33             }
34             if (document.form1.fechanacimiento.value=="")
35             {
36                 alert("Falta fecha de nacimiento");
37                 document.form1.fechanacimiento.focus();
38                 sw=1;
39                 return (false);
40             }
41             if (document.form1.Foto.value=="")
42             {
43                 alert("Falta seleccionar la imagen");
44                 document.form1.Foto.focus();
45                 sw=1;
46                 return (false);
47             }
48         }
```

Ilustración 27- Fragmento de código JavaScript

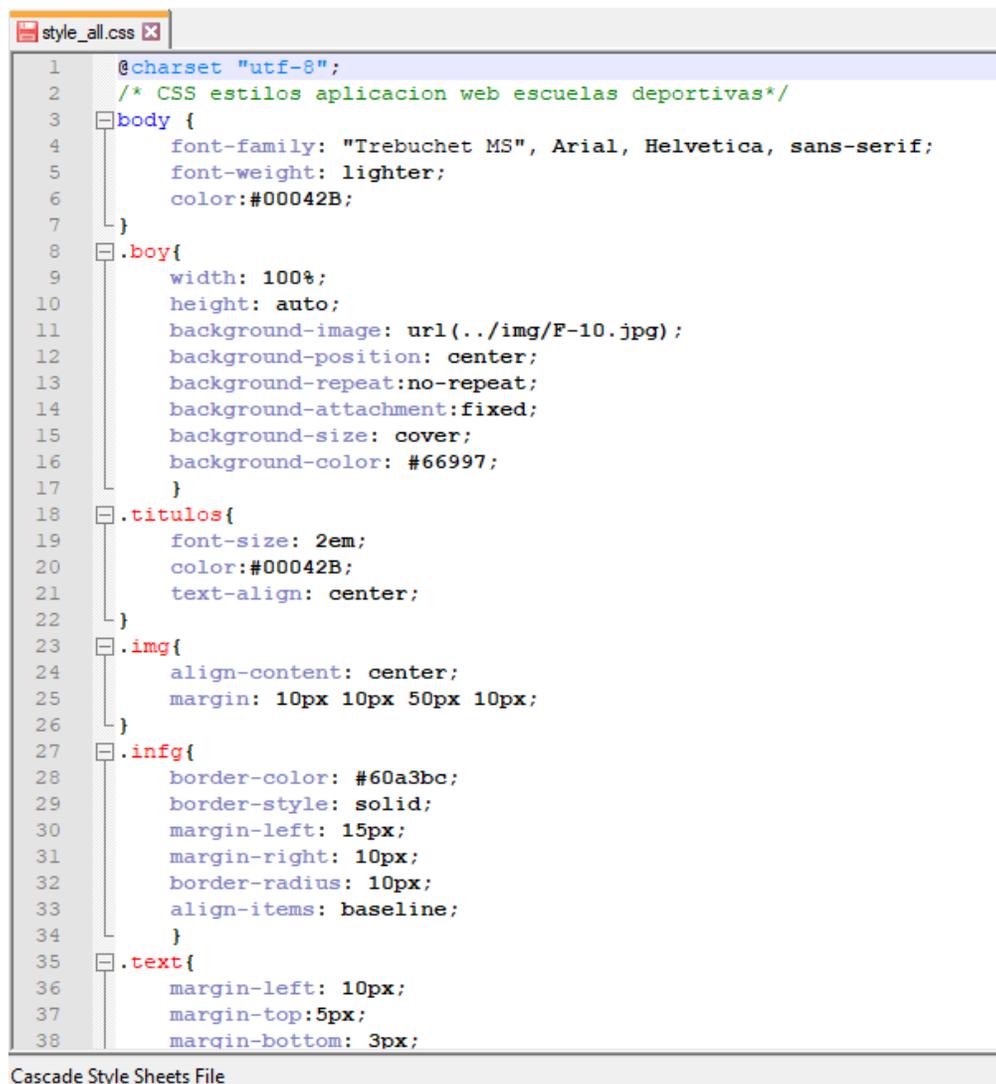
FUENTE: PROPIA

CSS

Las Hojas de estilo en cascada (del inglés *Cascading Stylesheets* CSS) es la siguiente tecnología que aprenderemos después de HTML. Mientras que HTML se utiliza para definir la estructura y la semántica del contenido, CSS se usa para darle estilo y posicionarlo visualmente. CSS se puede usar, por ejemplo, para cambiar la fuente, el color, el tamaño y el espaciado del contenido, para formar múltiples columnas, añadir animaciones y otros elementos decorativos.

Estas hojas de estilos nos permitirán diseñar la aplicación y mejorar la apariencia estética para el usuario; proporcionando colores, márgenes y formatos de textos que sean agradables a la vista de los distintos usuarios que entran interacción.

En el siguiente fragmento de codificación de hojas de estilos las cuales son utilizadas en la mayor parte de la aplicación Web:



```
1 @charset "utf-8";
2 /* CSS estilos aplicacion web escuelas deportivas*/
3 body {
4     font-family: "Trebuchet MS", Arial, Helvetica, sans-serif;
5     font-weight: lighter;
6     color:#00042B;
7 }
8 .boy{
9     width: 100%;
10    height: auto;
11    background-image: url(../img/F-10.jpg);
12    background-position: center;
13    background-repeat:no-repeat;
14    background-attachment:fixed;
15    background-size: cover;
16    background-color: #66997;
17 }
18 .titulos{
19     font-size: 2em;
20     color:#00042B;
21     text-align: center;
22 }
23 .img{
24     align-content: center;
25     margin: 10px 10px 50px 10px;
26 }
27 .infg{
28     border-color: #60a3bc;
29     border-style: solid;
30     margin-left: 15px;
31     margin-right: 10px;
32     border-radius: 10px;
33     align-items: baseline;
34 }
35 .text{
36     margin-left: 10px;
37     margin-top:5px;
38     margin-bottom: 3px;
```

Cascade Style Sheets File

Ilustración 28- Fragmento de código CSS

FUENTE: PROPIA

Fase de pruebas

Las pruebas realizadas a lo largo del desarrollo del proyecto están enfocadas a mejorar la calidad del software, probando cada opción por parte del usuario final evaluando la lógica de la aplicación logrando probar tanto los componentes como la funcionalidad del sistema; corrigiendo así posibles errores que estén presentes en el desarrollo de la aplicación web.

Para iniciar con las pruebas, se instaló una máquina virtual en la cual se instalarán todos los componentes para crear la aplicación, este servidor web proporcionado por el Colegio; cuenta con un sistema operativo Windows Server 2016 con las siguientes características:

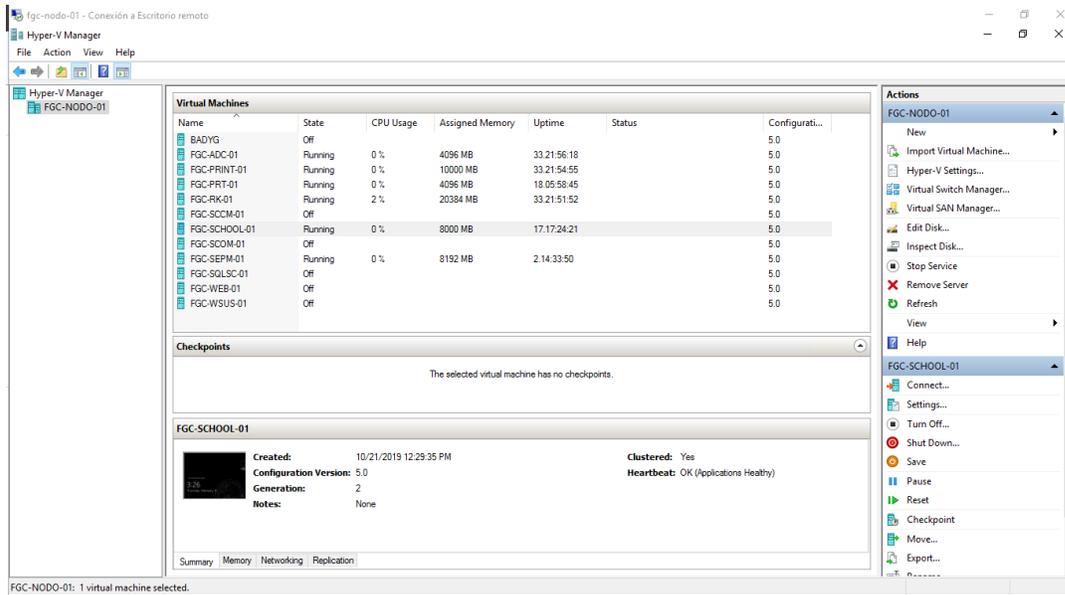


Ilustración 29- Instalación servidor

FUENTE: PROPIA

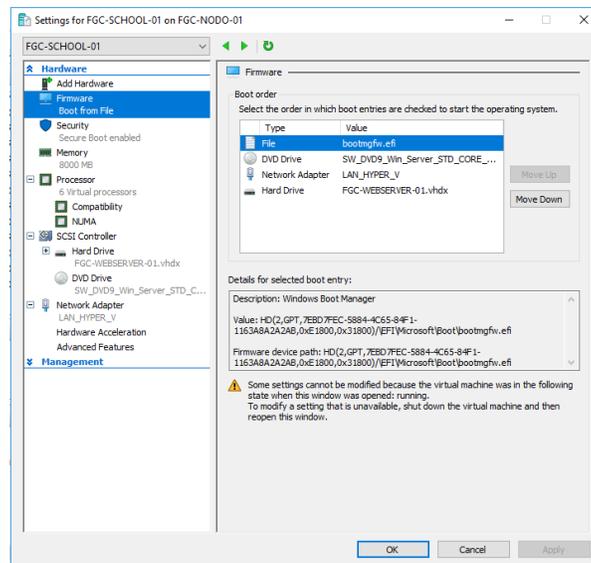


Ilustración 30- Configuración máquina virtual

FUENTE: PROPIA

Seguido de la configuración del servidor, se instala una distribución de servidor web Apache gratuita llamada XAMPP el cual va a ser nuestro intérprete para lenguaje de script PHP y sistema de gestión de base de datos MYSQL.

Una vez cargada la aplicación, se procede a validar su correcto funcionamiento, para lo cual se procede a confirmar el estado del servicio y los puestos TCP/IP específicos.

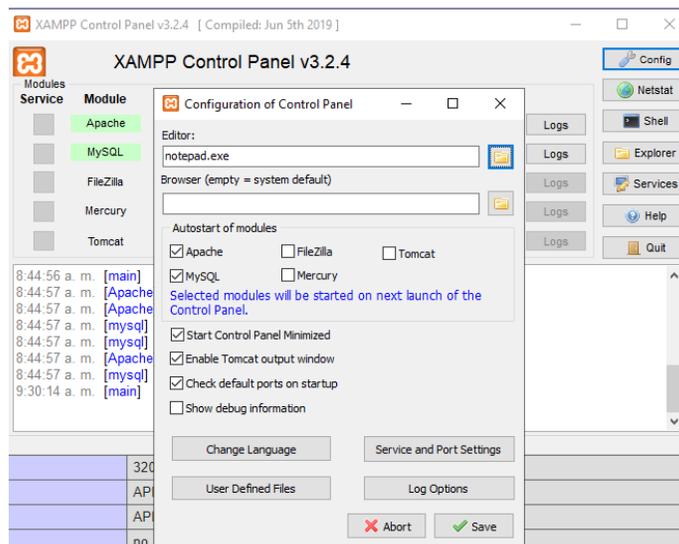


Ilustración 31- Instalación servicio servidor Web

FUENTE: PROPIA

PHP Version 7.3.9	
System	Windows NT GC-W10-CYESPITI 10.0 build 19042 (Windows 10) AMD64
Build Date	Aug 28 2019 09:21:06
Compiler	MSVC15 (Visual C++ 2017)
Architecture	x64
Configure Command	cmd /c "n /nologo configure.js "--enable-snapshot-build"--enable-debug-pack"--with-pdo-oci=c:\php-snap-build\deps_aux\oracle\x64\instantclient_12_1sdk_shared"--with-oci8-12c=c:\php-snap-build\deps_aux\oracle\x64\instantclient_12_1sdk_shared"--enable-object-out-dir=.obj"--enable-com-dotnet=shared"--without-analyzer"--with-pgo"
Server API	Apache 2.0 Handler
Virtual Directory Support	enabled
Configuration File (php.ini) Path	C:\WINDOWS
Loaded Configuration File	C:\xampp\php\php.ini
Scan this dir for additional .ini files	(none)
Additional .ini files parsed	(none)
PHP API	20180731
PHP Extension	20180731
Zend Extension	320180731
Zend Extension Build	API320180731.TS.VC15
PHP Extension Build	API20180731.TS.VC15
Debug Build	no
Thread Safety	enabled
Thread API	Windows Threads
Zend Signal Handling	disabled
Zend Memory Manager	enabled
Zend Multibyte Support	provided by mbstring
IPv6 Support	enabled
DTrace Support	disabled
Registered PHP Streams	php, file, glob, data, http, ftp, zip, compress.zlib, compress.bzip2, https, ftps, phar

Ilustración 32- Verificación de servicios Web

FUENTE: PROPIA

En la ubicación de archivos del servidor web se traslada el código fuente que se desarrolló anteriormente, para realizar las siguientes pruebas:

- ✚ Prueba de acceso e inicio de aplicación web.
- ✚ Prueba de inicio de sesión.
- ✚ Prueba de gestión de usuarios.
- ✚ Prueba de gestión de las escuelas deportivas.
- ✚ Prueba de gestión de los descriptores de logro.
- ✚ Prueba de gestión de estudiantes.
- ✚ Prueba de inscripción de un estudiante en escuela deportiva.
- ✚ Prueba de calificación de estudiante.

Con la siguiente tabla se describirá cada prueba a realizar quedando documentada cada operación o paso a seguir arrojando resultados, números de intentos para lograr el éxito de la operación, observaciones a considerar y las conclusiones de la prueba realizada.

Nombre de la Prueba

Nro.	Descripción de la operación	Resultado de la operación	Intentos	Éxito

Conclusiones de la prueba:

Tabla 35 - Formato para pruebas del sistema

Prueba de acceso e inicio de aplicación web.

En la siguiente tabla se detallan las operaciones que se consideran a evaluar del acceso e inicio de la aplicación web

Nombre de la Prueba Acceso e inicio de aplicación web

Nro.	Descripción de la operación	Resultado de la operación	Intentos	Éxito
1	Ingreso a dirección web para acceder al servidor y ubicación de archivos web.	Al ingresar a la dirección: http://localhost/dashboard/ Se evidencia que el servidor web funciona correctamente sin retrasos.	1	si
2	Ingreso a la dirección del aplicativo web	Al ingresar a la dirección: http://localhost/escuelas/ se evidencia que lo dirige correctamente al formulario de inicio de sesión correspondiente	1	SI

Conclusiones de la prueba: El servidor cumple con las características establecidas y no presenta atraso en la información suministrada.

Tabla 36- Prueba: acceso e inicio de aplicación web

Prueba de inicio de sesión.

En la siguiente tabla se detallan las operaciones que se consideran a evaluar del inicio de sesión por parte de un usuario.

Nombre de la Prueba		Inicio de sesión		
Nro.	Descripción de la operación	Resultado de la operación	Intentos	Éxito
1	Ingreso a la dirección del aplicativo web	Al ingresar a la dirección: http://localhost/escuelas/ se evidencia que lo dirige correctamente al formulario de inicio de sesión correspondiente	1	SI
2	Validar datos vacíos en el formulario de inicio de sesión	Muestra mensaje de advertencia para completar un valor en cada campo	1	SI
3	Validar usuario inexistente y contraseña errónea	Emite un mensaje al usuario el cual no está registrado o está inactivo para iniciar sesión	2	SI
4	Validar usuario registrado con contraseña incorrecta	Emite un mensaje al usuario de la contraseña ingresada es incorrecta	1	SI
5	Validar usuario registrado y contraseña correcta	La aplicación recibe los parámetros y lo dirige correctamente al menú de opciones que tiene asignadas a su perfil.	1	SI

Conclusiones de la prueba: El Proceso de validación de usuarios es correcto, los tiempos de los mensajes que se emiten son adecuados.

Tabla 37- Prueba: inicio de sesión

Prueba de gestión de usuarios.

En la siguiente tabla se detallan las operaciones que se consideran a evaluar frente a la creación, consulta, actualización, y eliminación de los datos un usuario para la aplicación web.

Nombre de la Prueba		gestión de usuarios		
Nr o.	Descripción de la operación	Resultado de la operación	Intentos	Éxito
1	Ingreso a la opción de administración y al formulario de Usuario nuevo	se evidencia que lo dirige correctamente al formulario correspondiente	1	SI
2	Validar datos vacíos en el formulario	Muestra correctamente el mensaje de advertencia para completar el valor	1	SI

		correspondiente en cada campo solicitado		
3	Validar usuario existente	Emite un mensaje al usuario el cual le permite saber que ese nombre de usuario ya está siendo utilizado.	1	SI
4	Validar usuario registrado	Emite un mensaje al usuario de la creación correcta del nuevo usuario y se evidencia en la base de datos el nuevo registro	1	SI
5	Seleccionar la opción listado de usuarios	La aplicación muestra la opción de búsqueda de usuarios agrupándolos por el perfil correspondiente	1	SI
6	Seleccionar la opción actualizar usuario	La aplicación muestra el formulario correspondiente desplegando la opción para seleccionar el usuario a ser modificado	2	SI
7	Seleccionar la opción cambiar clave	La aplicación muestra el respectivo formulario para cambiar la contraseña del usuario que se encuentra autenticado.	1	SI

Conclusiones de la prueba: El Proceso de gestión de usuarios es correcto, los tiempos de los mensajes que se emiten son adecuados y cumple a cabalidad con sus propósitos

Tabla 38- Prueba: Gestión de usuarios

Prueba de gestión de las escuelas deportivas.

En la siguiente tabla se detallan las operaciones que se consideran a evaluar de la gestión de las escuelas deportivas para la aplicación web.

Nombre de la Prueba		gestión escuelas deportivas		
Nr o.	Descripción de la operación	Resultado de la operación	Intentos	Éxito
1	Ingreso a la opción de escuelas la cual lo lleva al formulario de ingresar escuela	se evidencia que lo dirige correctamente al formulario correspondiente	1	SI
2	Validar datos vacíos en el formulario	Muestra correctamente el mensaje de advertencia para completar el valor correspondiente en cada campo solicitado	1	SI
3	Validación de escuela existente	Emite un mensaje al usuario el cual le permite saber que el valor ingresado ya está siendo utilizado.	1	SI
4	Validar creación de escuela	Emite un mensaje al usuario de la correcta creación de la escuela	1	SI

		deportiva y se evidencia en la base de datos el nuevo registro		
5	Seleccionar la opción listado de escuelas	La aplicación lo dirige correctamente al formulario correspondiente, mostrando el nombre y estado de las escuelas que se encuentran en la base de datos.	1	SI
6	Seleccionar la opción de actualizar escuela	La aplicación muestra el formulario correspondiente desplegando la opción para seleccionar de una lista desplegable la escuela a ser modificada	2	SI
Conclusiones de la prueba:		El Proceso de validación de creación de escuela deportiva es correcto, los tiempos de los mensajes que se emiten son adecuados.		

Tabla 39- Prueba: Gestión escuelas deportivas

Prueba de gestión descriptores de logro.

En la siguiente tabla se detallan las operaciones que se consideran a evaluar de la gestión de los descriptores de logro asociados a las escuelas deportivas para la aplicación web.

Nombre de la Prueba		gestión descriptores de logro		
Nr o.	Descripción de la operación	Resultado de la operación	Intentos	Éxito
1	Ingreso a la opción de descriptores la cual lo lleva al formulario	se evidencia que lo dirige correctamente al formulario correspondiente para añadir un nuevo descriptor de logro asociándolo correctamente a una escuela existente	1	SI
2	Validar datos vacíos en el formulario	Muestra correctamente el mensaje de advertencia para completar el valor correspondiente en cada campo solicitado	1	SI
3	Validar creación de descriptor	Emite un mensaje al usuario de la correcta creación del descriptor y se evidencia en la base de datos el nuevo registro	1	SI
5	Seleccionar la opción listado de descriptores	La aplicación lo dirige al formulario correspondiente, exponiendo un filtro para la búsqueda por escuela deportiva o todas; la búsqueda arroja texto del descriptor de logro, la fecha de creación y estado del mismo tal como se encuentran registrados en la base de datos.	3	SI

6	Seleccionar la opción de actualizar descriptor	La aplicación lo dirige al formulario correspondiente desplegando la opción para seleccionar de una lista desplegable la escuela, inmediatamente se habilita otra lista desplegable con parte del texto del descriptor a actualizar.	2	SI
Conclusiones de la prueba:		El Proceso de validación de creación de escuela deportiva es correcto, los tiempos de los mensajes que se emiten son adecuados.		

Tabla 40- Prueba: gestión descriptores de logro

Prueba de gestión de estudiantes.

En la siguiente tabla se detallan las operaciones que se consideran a evaluar de la gestión de los datos de estudiantes para la aplicación web.

Nombre de la Prueba gestión de estudiantes				
N°	Descripción de la operación	Resultado de la operación	Intentos	Éxito
1	Ingreso a la opción de estudiantes la cual lo lleva al formulario	se evidencia que lo dirige correctamente al formulario correspondiente para añadir un nuevo estudiante	1	SI
2	Validar datos vacíos en el formulario	Muestra correctamente el mensaje de advertencia para completar el valor correspondiente en cada campo solicitado	1	SI
3	Validar creación del estudiante	Emite un mensaje al usuario de la correcta creación del estudiante y se evidencia en la base de datos el nuevo registro	1	SI
5	Seleccionar la opción listado de estudiantes	La aplicación lo dirige al formulario correspondiente, exponiendo un filtro para la búsqueda por escuela deportiva o todas; la búsqueda arroja el nombre del estudiante, curso, fecha de nacimiento, foto y estado del mismo tal como se encuentran registrados en la base de datos.	1	SI
6	Seleccionar la opción de actualizar datos del estudiante	La aplicación lo dirige al formulario correspondiente desplegando la opción para buscar el estudiante escribiendo en un cuadro de texto parte del nombre, inmediatamente se despliega lista con los posibles	2	SI

	estudiantes; seleccionando un estudiante, se abre una ventana emergente con los datos pre cargados en la base de datos y así poder actualizar la información correspondiente. Al momento de guardar se valida y genera mensaje al usuario.		
--	--	--	--

Conclusiones de la prueba: El Proceso de validación de creación de escuela deportiva es correcto, los tiempos de los mensajes que se emiten son adecuados.

Tabla 41- Prueba: Gestión de estudiantes

Prueba de inscripción de un estudiante en escuela deportiva.

En la siguiente tabla se detallan las operaciones que se consideran a evaluar en la inscripción de un estudiante en escuela deportiva para la aplicación web.

Nombre de la Prueba		inscripción de un estudiante en escuela deportiva		
N°	Descripción de la operación	Resultado de la operación	Intentos	Éxito
1	Ingreso a la opción de estudiantes la cual lo lleva al formulario de inscribir estudiante	se evidencia que lo dirige correctamente al formulario correspondiente	1	SI
2	Validar datos vacíos en el formulario	Muestra correctamente el mensaje de advertencia para completar el valor correspondiente en cada campo solicitado	1	SI
3	Validación de estudiante existente	Emite un mensaje al usuario el cual le permite saber que el valor ingresado no se encuentra.	1	SI
4	Validar búsqueda de estudiante	Se busca el estudiante escribiendo en un cuadro de texto parte del nombre, inmediatamente se despliega lista con los posibles estudiantes y las escuelas deportivas que se encuentran activas en la base de datos	1	SI
5	Inscribir del listado de escuelas	Con el estudiante seleccionado puede escoger las escuelas a las que desee inscribir el estudiante; Al momento de guardar se valida y genera mensaje al usuario; se evidencia en la base de datos el nuevo registro	1	SI

Conclusiones de la prueba: El Proceso de validación de creación de escuela deportiva es correcto, los tiempos de los mensajes que se emiten son adecuados.

Tabla 42- Prueba: inscripción de un estudiante

Prueba de calificación estudiante.

En la siguiente tabla se detallan las operaciones que se consideran a evaluar de la calificación de un estudiante para la aplicación web.

Nombre de la Prueba		Calificación de un estudiante		
N°	Descripción de la operación	Resultado de la operación	Intentos	Éxito
1	Ingreso a la opción de escuelas la cual lo lleva al formulario de Calificar estudiantes.	se evidencia que lo dirige correctamente al formulario correspondiente	1	SI
2	Seleccionar escuela deportiva	De la lista desplegable se puede escoger la escuela que se desea calificar, validando que exista en la base de datos y que posea estudiantes inscritos.	1	SI
3	Listado de estudiantes inscritos	Después de escoger la escuela deportiva se listan en una tabla los estudiantes que tienen inscripción correcta y con calificación pendiente para realizar la acción requerida, de la cual puede escoger un estudiante a la vez.	2	SI
4	Calificar estudiante seleccionado	La aplicación lo dirige correctamente al formulario correspondiente, mostrando los datos del estudiante, el nombre de la escuela, los descriptores de logro asociados a la misma y las tres opciones de calificación.	1	SI
6	Guardar calificación	Verificación de datos vacíos y los dos campos adicionales de aspectos positivos y aspectos por mejorar se validan todos los campos y se guarda la información en la base de datos, se evidencia en la base de datos el nuevo registro y se muestra el mensaje respectivo al usuario de la correcta calificación.	3	SI
Conclusiones de la prueba:		El Proceso de validación de creación de escuela deportiva es correcto, los tiempos de los mensajes que se emiten son adecuados.		

Tabla 43- Prueba: Calificación de un estudiante

Fase de implementación

Para liberar la aplicación web, se instaló una máquina virtual en la cual se instalarán todos los componentes para crear la aplicación, este servidor web proporcionado por el Colegio; cuenta con un sistema operativo Windows Server 2016 con las siguientes características:

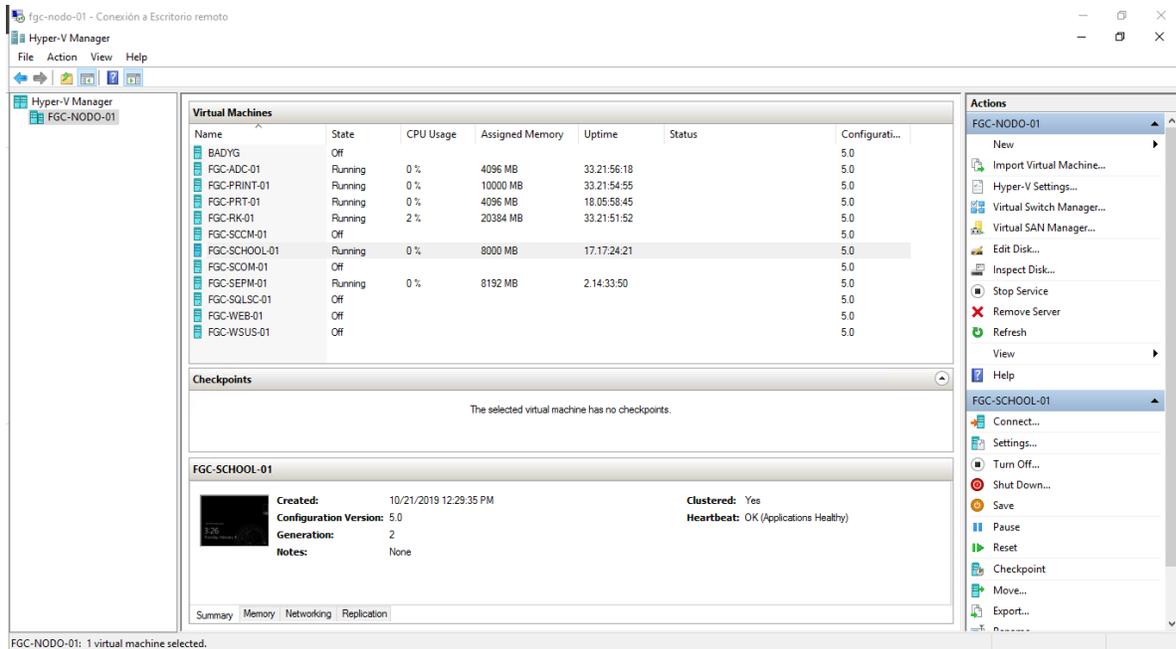


Ilustración 33- Instalación servidor implementación

FUENTE: PROPIA

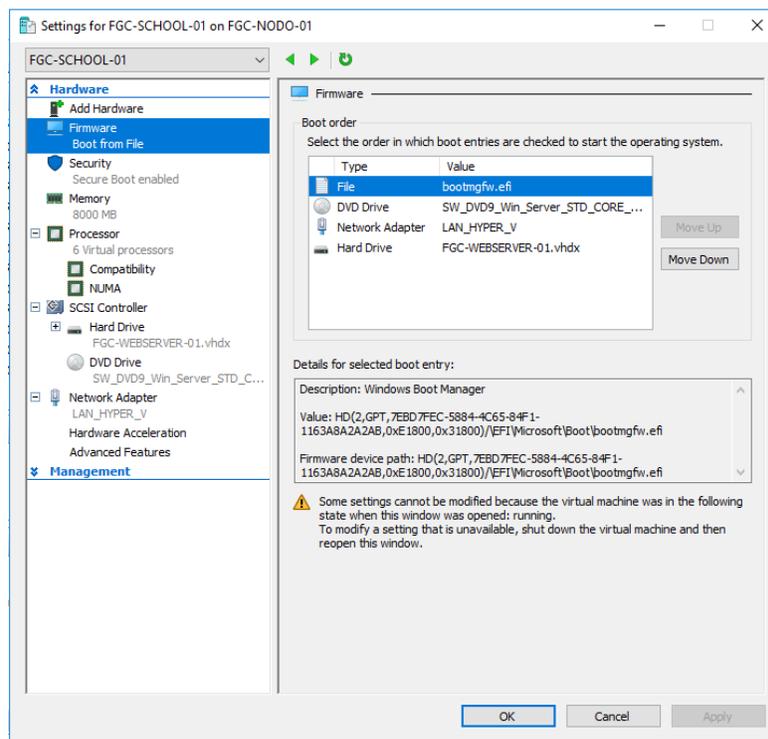


Ilustración 34- Configuración máquina virtual implementación

FUENTE: PROPIA

Seguido de la configuración del servidor, se instala una distribución de servidor web Apache gratuita llamada XAMPP el cual va a ser nuestro intérprete para lenguaje de script PHP y sistema de gestión de base de datos MYSQL.

Una vez cargada la aplicación, se procede a validar su correcto funcionamiento, para lo cual se procede a confirmar el estado del servicio y los puestos TCP/IP específicos.

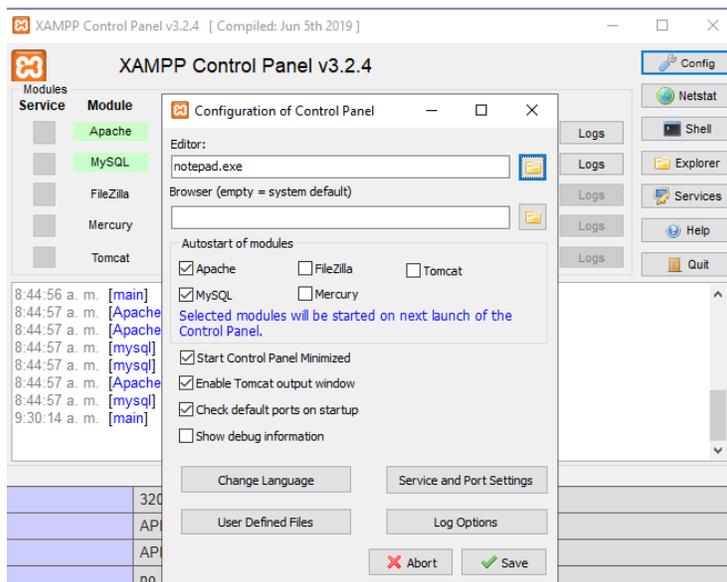


Ilustración 35- Servicios Web implementación

FUENTE: PROPIA

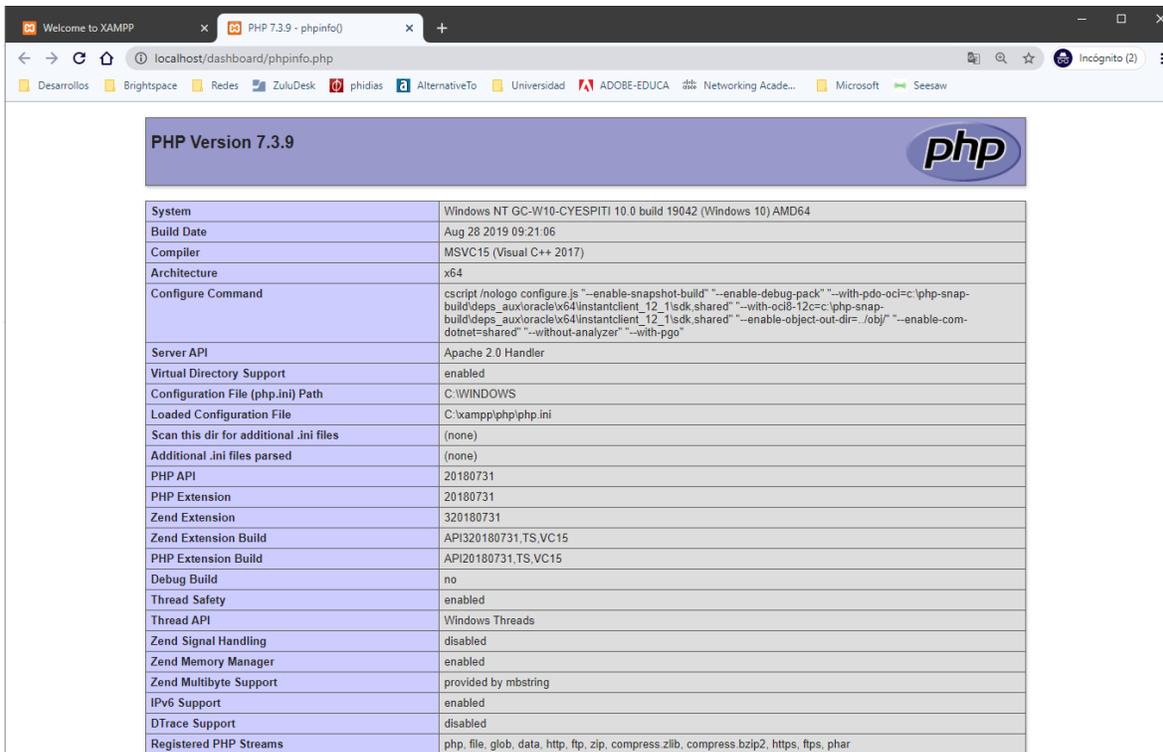


Ilustración 36- Verificación servidor Web implementación

FUENTE: PROPIA

En la ubicación predeterminada del servidor web se traslada el código fuente que se desarrolló anteriormente y se realiza la limpieza de los datos de prueba para su liberación y funcionamiento. Quedando con datos base de escuelas, descriptores, estudiantes e inscripciones previas de estudiantes.

Tabla	Acción	Filas	Tipo	Cotejamiento	Tamaño	Residuo a depurar
<input type="checkbox"/> amodulos	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	187	InnoDB	utf8_spanish2_ci	48 KB	-
<input type="checkbox"/> aspectos	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	0	InnoDB	latin1_swedish_ci	64 KB	-
<input type="checkbox"/> descriptores	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	270	InnoDB	latin1_swedish_ci	80 KB	-
<input type="checkbox"/> escuelasdep	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	56	InnoDB	latin1_swedish_ci	16 KB	-
<input type="checkbox"/> estudescriptores	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	0	InnoDB	utf8_spanish2_ci	32 KB	-
<input type="checkbox"/> estudiantes	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	493	InnoDB	latin1_swedish_ci	96 KB	-
<input type="checkbox"/> inscripciones	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	100	InnoDB	latin1_swedish_ci	48 KB	-
<input type="checkbox"/> menus	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	4	InnoDB	utf8_spanish2_ci	16 KB	-
<input type="checkbox"/> modulos	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	20	InnoDB	utf8_spanish2_ci	48 KB	-
<input type="checkbox"/> usuariosescuela	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	8	InnoDB	latin1_swedish_ci	48 KB	-
<input type="checkbox"/> usuarios	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	12	InnoDB	latin1_swedish_ci	32 KB	-
11 tablas	Número de filas	1.070	InnoDB	utf8_spanish2_ci	528 KB	0 B

Ilustración 37- Base de datos limpia

FUENTE: PROPIA

Una vez cargada todos los archivos de la aplicación en el servidor, se procede a validar su correcto despliegue y funcionamiento, para lo cual se realiza el ingreso a la aplicación desde el servidor y funciona correctamente.

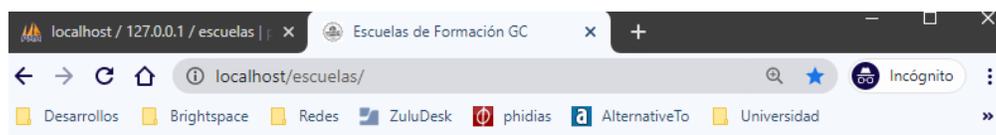


Ilustración 38- Estado aplicación web funcional

FUENTE: PROPIA

Fase mantenimiento y aseguramiento

En esta última fase, la aplicación que se ha diseñado e implementado cumple con los requerimientos establecidos, permitiendo tener una aplicación funcional. El mantenimiento de la aplicación está enfocado a la administración de los datos para que siempre se encuentren la mayor parte del tiempo actualizados, así poder tener información precisa y confiable.

Es importante que los archivos de la aplicación sigan la jerarquía establecida al momento de ser almacenados, ya que de no ser así no funcionará correctamente. Los archivos almacenados necesitan de otros para integrar funciones y acceder a ellos se utiliza la ruta explícita del archivo almacenado, de manera que si esta ruta cambia será imposible para la aplicación el correcto funcionamiento.

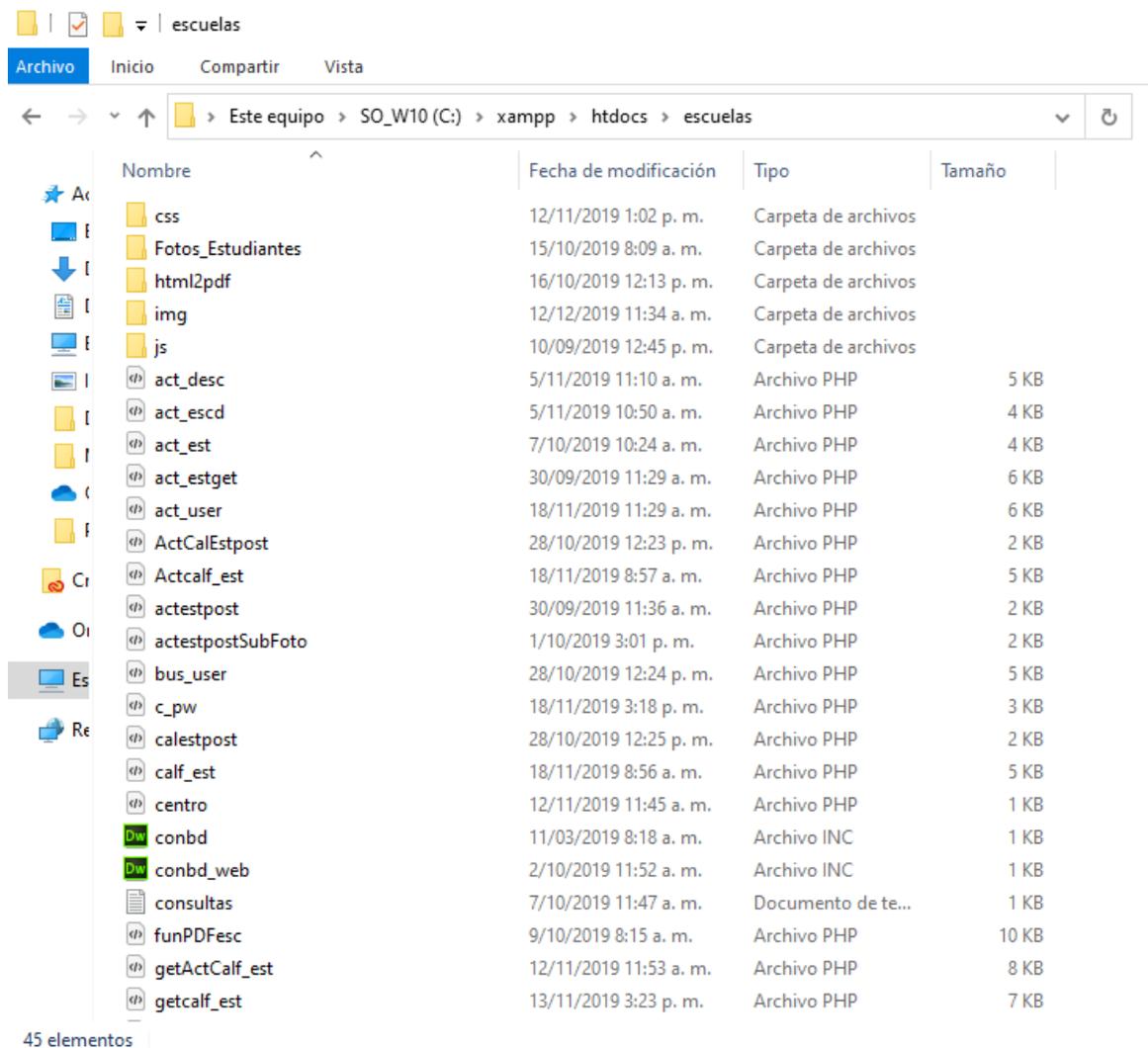


ILUSTRACIÓN 39- UBICACIÓN DE ARCHIVOS

FUENTE: PROPIA

Plan de calidad

El propósito de este plan de calidad es analizar las fases de la metodología propuesta asegurando la calidad del aplicativo web a desarrollar, teniendo como base los requerimientos explícitamente establecidos y documentados, para adoptar la confianza del usuario y satisfacer las necesidades expuestas.

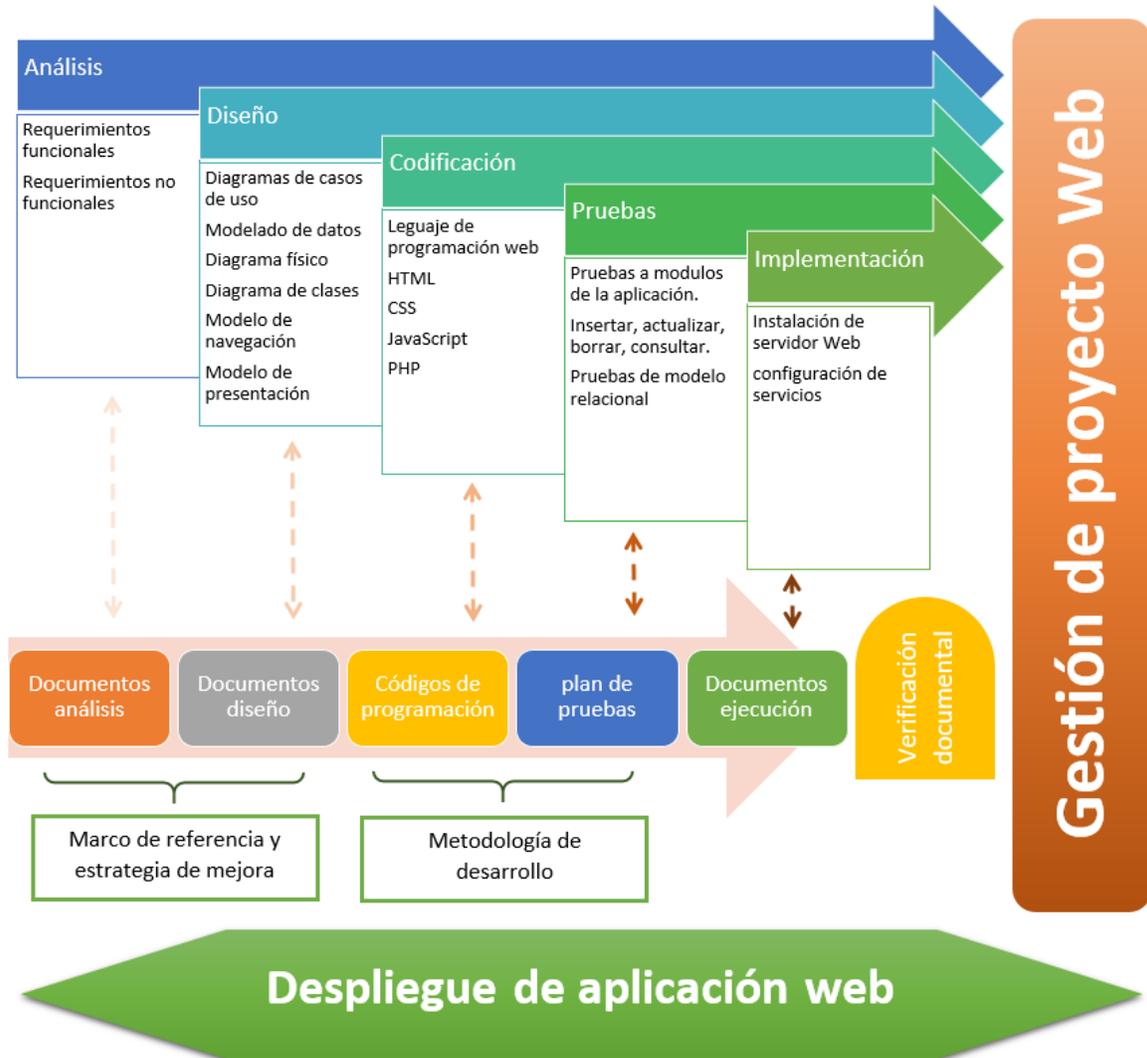


Ilustración 40- Plan de calidad

FUENTE: PROPIA

Nombre actividad	Resultado	Responsable	Observaciones
Acercamiento y levantamiento de información inicial del proyecto	Primer acercamiento y entender las necesidades que se tienen en el proceso.	Analista	Entender la necesidad del usuario para aplicar una solución efectiva
Análisis de requerimientos funcionales y no funcionales	Se obtiene el listado de los requerimientos de la aplicación web	Analista	Alcance de cada requerimiento
Diseño de diagramas de uso, físico, clases, navegación y presentación	Se logra entender el modelo abstracto de la aplicación web, aplicándolo a una metodología de desarrollo	Analista	Modelar de manera clara para que el programador entienda las necesidades del usuario
Programación de la aplicación web	Código fuente de las funciones solicitadas y los diseños para presentar la aplicación al usuario	Programador	Tener ordenado el código y dejan en claro que módulos y opciones se deben tener en específico
Pruebas a los diferentes opciones desarrolladas para dar solución a los requerimientos iniciales	Estas pruebas se realizaron en paralelo a la programación de la aplicación verificando los requerimientos y levantamiento de información inicial	Programador	Realizar pruebas y utilizar documentación para las pruebas realizadas
Implementación del servidor para desplegar aplicación web	Configuración del servidor, servicios web necesarios para la importación del código fuente desarrollado y probado para desplegar aplicación web	Analista y Programador	Realizar manual de usuario para que todos tengan los mismos criterios de trabajo

Tabla 44- Descripción plan de calidad

Matriz de riesgos

Probabilidad / impacto	Valor
Baja	1
Media	2
Alta	3

Nivel de riesgo	
Nivel	Valor
Alto	6 a 9
Medio	3 a 4
Bajo	1 a 2

#	Nombre del riesgo	Riesgo	Síntoma	Probabilidad	Impacto	Prioridad	Estrategia prevención	Responsable de la acción de respuesta
1	Servicio de internet ofrecido por el colegio a todos sus estudiantes, docentes y empleados.	Afectación temporal al servicio de internet ofrecido.	Suspensión temporal o total del servicio de internet principal contratado por el colegio	1	3	3	Contratación de un canal de internet secundario o de contingencia con un operador e infraestructura diferente	Departamento de sistemas.
2	Prestación de Servicios de infraestructura tecnológica	Interrupción en los de Servicios de infraestructura tecnológica	Situación en la cual por cualquier evento (Natural, accidental o premeditado) se cause la reconfiguración al 100%	1	3	3	Monitoreo constante y cronograma de mantenimientos establecidos	Departamento de sistemas.
3	Perdida de la información por daños de hardware y software del DataCenter	Pérdida total o parcial de datos sensibles de estudiantes y padres de familia	Incumplimiento de la entrega total de boletines a padres de familia, Con la información completa, en las condiciones esperadas	1	3	3	Migración total de servicios a ambientes CLOUD y backups cada 12 horas incrementales para restauración.	Departamento de sistemas.

Tabla 45- Matriz de riesgos

Conclusiones

En el progreso de este proyecto, el cual estaba orientado a que el Colegio tuviera una aplicación que ayudara a automatizar y agilizar el proceso de registro, calificación y envío de informes de los logros alcanzados por los estudiantes en las escuelas deportivas a los acudientes, queda demostrado lo viable y posible de utilizar herramientas gratuitas o de bajo costo para obtener resultados que cumplan con los objetivos propuestos.

Con el desarrollo de la aplicación los usuarios tenían toda la información para poder realizar sus tareas de calificación de los estudiantes, agilizando tiempos de respuesta a otras actividades pedagógicas del proceso formativo.

Haciendo uso de las tecnologías de información se eliminó el flujo de documentos Word y Excel que a menudo, estaban descontrolados con las resultantes dificultades para su ubicación generando retrasos en el proceso; se logró tener la mayor parte de la información centralizada haciéndola accesible para así tomar decisiones pedagógicas adecuadas. Esto también se vio reflejando en un impacto positivo en la comunidad, y ayudando al medio ambiente con el menor uso del papel impreso.

Bibliografía

- Cabello, E. (29 de 10 de 2013). *Trimestre V P.S.T. II*. Obtenido de <http://evangellyscarolinacabellorodriguez.blogspot.com/2013/10/fases-o-etapas-de-la-metodologia-uwe.html>
- Colegio Gimnasio Campestre. (15 de 08 de 2019). *Gimnasio Campestre - Escuelas de Formación*. Obtenido de <https://www.campestre.edu.co/phi/section/display/section=7739>
- EcuRed. (6 de 7 de 2019). *Metodologías de desarrollo de Software*. Obtenido de https://www.ecured.cu/Metodologias_de_desarrollo_de_Software
- edukativos. (06 de 05 de 2016). *Desarrollo de Sistemas de Jackson*. Obtenido de <https://www.edukativos.com/apuntes/archives/10577>
- esacademic.com. (2010). *Metodología de desarrollo de software*. Obtenido de https://esacademic.com/dic.nsf/eswiki/801366#Rapid_Application_Development_.28RAD.29
- inteldig.com. (2018). *Tecnologías de Información: Sistemas de Gestión y Administración*. Obtenido de Modelado de Datos: Definición, Usos y Tipos: <https://www.tecnologias-informacion.com/modeladodatos.html>
- INTERNET YA. (14 de 11 de 2016). *Ventajas y beneficios de las aplicaciones Web*. Obtenido de <https://www.internetya.co/ventajas-y-beneficios-de-las-aplicaciones-web/>
- Koch, N. (10 de 8 de 2016). *UWE – UML-based Web Engineering*. Obtenido de <http://uwe.pst.ifi.lmu.de/teachingTutorialSpanish.html>
- Nora Koch, Marianne Busch. (10 de 08 de 2016). *UWE - UML-based web engineering*. Obtenido de <http://uwe.pst.ifi.lmu.de/index.html>
- Peter Cowburn. (28 de 10 de 2019). *¿Qué es PHP?* Obtenido de <https://www.php.net/>
- proyectosagiles.org. (2008). *Qué es SCRUM – Proyectos Ágiles*. Obtenido de <https://proyectosagiles.org/que-es-scrum/>
- Wikipedia. (28 de 10 de 2019). *Wikipedia*. Obtenido de https://es.wikipedia.org/wiki/Lenguaje_de_programaci%C3%B3n
- Developer Mozilla Org. (8 de 8 de 2020). *¿Qué es JavaScript?* Obtenido de https://developer.mozilla.org/es/docs/Learn/JavaScript/First_steps/Qu%C3%A9_es_JavaScrip
- Developer Mozilla org. (10 de 12 de 2020). Obtenido de HTML: Lenguaje de etiquetas de hipertexto: <https://developer.mozilla.org/es/docs/Web/HTML>

Php.net. (2021). *Documentación PHP*. Obtenido de ¿Qué es PHP?:
<https://www.php.net/manual/es/intro-what-is.php>