



# UTMACH

FACULTAD DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

DESARROLLO DE APLICACIÓN WEB PARA GESTIÓN  
ADMINISTRATIVA DE ACADEMIA DE TAEKWONDO UTILIZANDO  
FRAMEWORK REACT NATIVE Y ENTORNO DE EJECUCIÓN NODEJS

AREVALO MIRANDA ERICK FABRICIO  
INGENIERO DE SISTEMAS

MACHALA  
2022



# UTMACH

FACULTAD DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

DESARROLLO DE APLICACIÓN WEB PARA GESTIÓN  
ADMINISTRATIVA DE ACADEMIA DE TAEKWONDO  
UTILIZANDO FRAMEWORK REACT NATIVE Y ENTORNO DE  
EJECUCIÓN NODEJS

AREVALO MIRANDA ERICK FABRICIO  
INGENIERO DE SISTEMAS

MACHALA  
2022



# UTMACH

FACULTAD DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

TRABAJO TITULACIÓN  
PROPUESTAS TECNOLÓGICAS

DESARROLLO DE APLICACIÓN WEB PARA GESTIÓN ADMINISTRATIVA DE  
ACADEMIA DE TAEKWONDO UTILIZANDO FRAMEWORK REACT NATIVE Y  
ENTORNO DE EJECUCIÓN NODEJS

AREVALO MIRANDA ERICK FABRICIO  
INGENIERO DE SISTEMAS

HONORES TAPIA JOOFRE ANTONIO

MACHALA, 20 DE SEPTIEMBRE DE 2022

MACHALA  
2022

# Tesis de Erick Arévalo

## INFORME DE ORIGINALIDAD

3%

INDICE DE SIMILITUD

4%

FUENTES DE INTERNET

0%

PUBLICACIONES

2%

TRABAJOS DEL ESTUDIANTE

## FUENTES PRIMARIAS

1	<a href="http://repositorio.utmachala.edu.ec">repositorio.utmachala.edu.ec</a> Fuente de Internet	1%
2	Submitted to Universidad Técnica de Machala Trabajo del estudiante	1%
3	<a href="http://dspace.ups.edu.ec">dspace.ups.edu.ec</a> Fuente de Internet	1%

Excluir citas

Activo

Excluir coincidencias < 1%

Excluir bibliografía

Activo

## CLÁUSULA DE CESIÓN DE DERECHO DE PUBLICACIÓN EN EL REPOSITORIO DIGITAL INSTITUCIONAL

El que suscribe, AREVALO MIRANDA ERICK FABRICIO, en calidad de autor del siguiente trabajo escrito titulado DESARROLLO DE APLICACIÓN WEB PARA GESTIÓN ADMINISTRATIVA DE ACADEMIA DE TAEKWONDO UTILIZANDO FRAMEWORK REACT NATIVE Y ENTORNO DE EJECUCIÓN NODEJS, otorga a la Universidad Técnica de Machala, de forma gratuita y no exclusiva, los derechos de reproducción, distribución y comunicación pública de la obra, que constituye un trabajo de autoría propia, sobre la cual tiene potestad para otorgar los derechos contenidos en esta licencia.

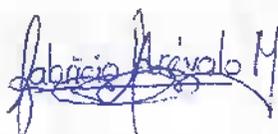
El autor declara que el contenido que se publicará es de carácter académico y se enmarca en las disposiciones definidas por la Universidad Técnica de Machala.

Se autoriza a transformar la obra, únicamente cuando sea necesario, y a realizar las adaptaciones pertinentes para permitir su preservación, distribución y publicación en el Repositorio Digital Institucional de la Universidad Técnica de Machala.

El autor como garante de la autoría de la obra y en relación a la misma, declara que la universidad se encuentra libre de todo tipo de responsabilidad sobre el contenido de la obra y que asume la responsabilidad frente a cualquier reclamo o demanda por parte de terceros de manera exclusiva.

Aceptando esta licencia, se cede a la Universidad Técnica de Machala el derecho exclusivo de archivar, reproducir, convertir, comunicar y/o distribuir la obra mundialmente en formato electrónico y digital a través de su Repositorio Digital Institucional, siempre y cuando no se lo haga para obtener beneficio económico.

Machala, 20 de septiembre de 2022



AREVALO MIRANDA ERICK FABRICIO  
0705289205

## **DEDICATORIA**

El presente trabajo de titulación se lo dedico a Dios por ayudarme a seguir adelante a pesar de todos los obstáculos que he tenido que sobrellevar a lo largo de mi carrera universitaria.

A mi madre Sandra Miranda, quien siempre ha sido mi motivación, ejemplo de superación e inspiración para no rendirme y cumplir cada una de las metas que me proponga.

A mi abuelito Teodoro Miranda, quien en vida fue como un padre para mí, hombre honrado, fuerte, responsable y lleno de cariño para todos sus hijos, nietos y familiares, dándome siempre consejos y apoyo en todo momento para que sea una persona responsable, lleno de valores y virtudes.

A mi abuelita Chanena Monserrate, quien ha sido como una segunda madre para mí, a mi tía Ana Miranda y mi primita Juliana Miranda quienes son mi causa de ser mejor cada día.

Y Finalmente a mis tíos, primos, primas que se encuentran en otro país, amigos, conocidos y a mi querida enamorada, quienes siempre me han estado apoyando y dando su granito de arena para hacer posible este momento lleno de alegría.

Sr. Erick Fabricio Arévalo Miranda

## **AGRADECIMIENTO**

Agradezco a Dios por permitirme culminar con esta hermosa etapa de mi vida estudiantil y principalmente a mi familia por su apoyo incondicional y por ser partícipes de este momento lleno de felicidad.

A mi enamorada y a mis amigos más cercanos por estar presentes en cada momento de mi vida y saber dar la mano cuando más los necesitaba.

De igual manera agradezco a mi prestigiosa Universidad Técnica de Machala por darme la oportunidad de formarme como profesional y conocer en el camino a grandes y talentosos maestros que me guiaron con sus enseñanzas y al Ing. Joffre Honores Tapia, tutor de la tesis quien, con su esfuerzo y dedicación, supo brindarme sus conocimientos para culminar con éxito mis estudios.

Sr. Erick Fabricio Arévalo Miranda

## RESUMEN

Actualmente, se vive en un mundo globalizado y tecnológico con herramientas que permiten la intercomunicación entre las personas y la facilidad de tener cualquier tipo de información al alcance de nuestras manos; ante esta situación se ha incentivado la elaboración de sistemas que ayuden a mejorar la calidad de vida de la sociedad y aporten bienestar a la comunidad, ayudando de esta manera a la toma de decisiones.

En la presente época es común observar instrumentos tecnológicos inmersos en distintas áreas de la sociedad tales como: páginas web, aplicaciones móviles, redes sociales, dispositivos inteligentes, etc; estos instrumentos se encuentran presentes en diferentes áreas como educación, social, salud, cultural, empresarial, industrial y deportiva. Muchas empresas ven la necesidad de incorporar la tecnología para lograr establecer ventajas competitivas, alcanzar mayores niveles de productividad y crecimiento permitiendo lograr la optimización de los procesos de producción, organización, despachos, compra, venta, capacitación, entre otros.

En el ámbito deportivo los dispositivos tecnológicos son considerados de gran ayuda pues han permitido crear un nuevo capítulo en las competiciones, entrenamiento y apoyo a los deportistas dotándolos de herramientas para mejorar su preparación contando además con aplicaciones informáticas que permiten monitorear su rendimiento lo cual ha revolucionado todas las disciplinas deportivas.

El Taekwondo es un deporte olímpico de origen coreano considerado uno de los más ejercidos actualmente por niños, jóvenes y adultos cuyo propósito es mejorar la parte espiritual, física y marcial de quien lo practique; en la parte competitiva los atletas suelen prepararse de acuerdo a esquemas tradicionales y diversas formas de entrenamiento para lograr un nivel óptimo de competencia. Gracias a la adaptación tecnológica surgen nuevos métodos que permiten aumentar la preparación física y elástica del deportista, además de la integración de equipos tecnológicos como petos electrónicos, sistemas audiovisuales, pateadores con sensores, entre otros, que permiten innovar este deporte.

Con la aparición del internet, una de las herramientas tecnológicas que ha tenido mayor crecimiento son las aplicaciones Web, pues, son muy demandadas a nivel empresarial y laboral por su adaptabilidad, portabilidad, eficiencia y automatización

de procesos; además, permiten la elaboración de páginas personalizadas, la reproducción automática de contenidos, el desarrollo de comercio electrónico también conocido como e-commerce, entre otros.

“Nietos’ Academia” es una escuela de Taekwondo orientada al crecimiento deportivo de niños, niñas, jóvenes y adultos situada en la ciudad de Machala cuyos procesos como el registro de alumnos, el desarrollo de eventos de competencia y la creación de los pools o llaves de peleas se llevan a cabo de manera manual generando pérdidas de tiempo e inconformidad en los propietarios del establecimiento; frente a estos inconvenientes se plantea el uso de una herramienta tecnológica que permita satisfacer las necesidades de los involucrados. Es por ello que el presente trabajo de titulación denominado “Desarrollo de una aplicación web para la gestión administrativa de la academia de Taekwondo utilizando framework React Native y entorno de ejecución NodeJs” se elabora con la finalidad de automatizar los procesos administrativos de la escuela “Nieto’s Academia”.

Las tecnologías necesarias para la elaboración de la aplicación web son: un sistema gestor de base de datos NoSQL MongoDB, entorno de desarrollo IDE Visual Studio Code, para el Front-end el framework React Native por parte del cliente y para el Back-end el entorno de ejecución NodeJs por parte del servidor; todo este conjunto de herramientas permite realizar una plataforma dinámica, segura, adaptable, eficiente y de calidad.

El análisis de requisitos y las etapas de desarrollo se realizan con fundamentación en la metodología ágil “SWIRL” por su adaptabilidad para la elaboración de aplicaciones web; además, ofrece ventajas como la reducción de costos, la interacción con el usuario en cada momento y la modificación de los requisitos en cualquier etapa del proyecto. Finalmente, para la respectiva evaluación del sistema se sigue el estándar de calidad de software ISO/IEC-9126 garantizando la usabilidad, confiabilidad, eficiencia, funcionalidad, portabilidad y la facilidad de mantenimiento por parte de la plataforma.

**Palabras claves:** Front-end, Back-end, SWIRL, React, NodeJs, Academia, Gestión Administrativa, Aplicación web.

## **ABSTRACT**

Currently, we live in a globalized and technological world with tools that allow intercommunication between people and the ease of having any type of information at our fingertips; given this situation, the development of systems that improve the quality of life of society and provide well-being to the community has been encouraged, thus helping decision-making.

At the present time it is common to observe technological instruments immersed in different areas of society such as: web pages, mobile applications, social networks, smart devices, etc; these instruments are present in different areas such as education, social, health, cultural, business, industrial and sports. Many companies see the need to incorporate technology to establish competitive advantages, achieve higher levels of productivity and growth, allowing optimization of production processes, organization, dispatches, purchase, sale, training, among others.

In the sports field, technological devices are considered of great help because they have allowed the creation of a new chapter in competitions, training and support for athletes, providing them with tools to improve their preparation, also having computer applications that allow them to monitor their performance, which has revolutionized all sports disciplines.

Taekwondo is an Olympic sport of Korean origin considered once of the most currently exercised by children, youth and adults whose purpose is to improve the spiritual, physical and martial part of those who practice it; in the competitive part, athletes usually prepare according to traditional schemes and various forms of training to achieve an optimal level of competition. Thanks to technological adaptation, new methods arise that allow increasing the physical and elastic preparation of the athlete, in addition to the integration of technological equipment such as electronic bibs, audiovisual systems, kickers with sensors, among others, that allow innovating this sport.

With the appearance of the Internet, one of the technological tools that has had the greatest growth are web applications, since they are highly demanded at the business and labor level due to their adaptability, portability, efficiency and process automation; in addition, they allow the creation of personalized pages, the automatic

reproduction of content, the development of electronic commerce also known as e-commerce, among others.

“Nieto’s Academy” is a Taekwondo schools oriented to the sports growth of boys, girls, youth and adults located in the city of Machala whose processes such as student registration, the development of competition events and the creation of pools or keys of fights are carried out manually, generating waste of time and nonconformity in the owners of the establishment; faced with these inconveniences, the use of a technological tool that allows satisfying the needs of those involved is proposed. That is why the present degree work called “Development of a web application for the administrative management of the Taekwondo academy using React Native framework and NodeJs execution environment” is elaborated with the purpose of automating the administrative processes of the school “Nieto’s Academy”.

The necessary technologies for the development of the web application are: a NoSQL MongoDB database management system, IDE Visual Studio Code development environment, for the Front-end the React Native framework by the client and for the Back-end the NodeJs execution environment on the server side; all this set of tools allows for a dynamic, secure, adaptable, efficient and quality platform.

The analysis of requirements and the stages of development are carried out based on the agile methodology “SWIRL” due to its adaptability for the development of web applications; in addition, it offers advantages such as cost reduction, interaction with the user at all times and the modification of requirements at any stage of the project. Finally, for the respective evaluation of the system, the ISO/IEC-9126 software quality standard is followed, guaranteeing the usability, reliability, efficiency, functionality, portability and ease of maintenance by the platform.

**Keywords:** Front-end, Back-end, SWIRL, React, NodeJs, Academy, Administrative Management, Web Application.

## INTRODUCCIÓN

La tecnología está cambiando la realidad social contemporánea, eliminando las barreras del tiempo y distancia, permitiendo compartir información y mejorar procesos en diferentes ámbitos como educativo, social, salud, cultural, deportivo, empresarial, entre otros. La aparición de nuevas herramientas tecnológicas ha ayudado a solucionar problemas y satisfacer necesidades que surgen ante una creciente ola de demanda por parte de las personas; ante esta situación uno de los mejores aliados es el internet pues ha permitido interconectar a una sociedad globalizada y exigente; ofreciendo un sinnúmero de prestaciones y servicios que permiten la automatización segura y eficaz.

La introducción de internet ha facilitado la intercomunicación de todo el mundo, pues, permite producir, distribuir y utilizar datos digitalizados en cualquier formato y momento, todo lo mencionado puede hacerse posible gracias al desarrollo web, el cual ha producido un cambio notable en el ámbito empresarial, ya que ayuda a mejorar el aspecto de las empresas con la creación de aplicaciones web que permiten la optimización de procesos y la automatización de los mismos, la administración de servicios y la transformación digital de su información garantizando la seguridad de sus datos.

El desarrollo web evoluciona rápidamente dando como resultado la aparición de nuevos lenguajes de programación, tecnologías de software y metodologías que ayudan a la creación de aplicaciones web; frente a esto, la selección de herramientas de desarrollo adecuadas se vuelven imprescindibles para el proceso de elaboración de las mismas, permitiendo conseguir un sistema web funcional, usable, confiable, eficiente, mantenible y portable con métricas de calidad que cumplan con los estándares internacionales para la industria del software.

Las aplicaciones web se han convertido en un gran aliado de las personas por su adaptabilidad y usabilidad, básicamente se pueden utilizar desde una computadora hasta un dispositivo móvil lo que las convierte en una poderosa herramienta para cualquier tipo de necesidad que se tenga; pertenecen a la arquitectura cliente/servidor, es decir, un ordenador solicita un servicio (cliente) y otro está a la

espera de recibir las solicitudes y responderlas (servidor) [1], permitiendo de esta manera conectar a varios usuarios al mismo tiempo.

Por lo tanto, resulta útil la creación de una aplicación web, donde se mantenga la información almacenada, segura y ordenada con datos reales y actualizados sobre los procesos que se lleven a cabo dentro de la escuela de taekwondo “Nieto’s Academia” garantizando la comodidad y facilidad para sus propietarios.

“Nieto’s Academia” es una escuela de Taekwondo orientada al entrenamiento deportivo donde se puede aprender defensa personal, mejorar la condición física, aumentar la confianza y autoestima, acrecentar la capacidad de concentración, aprender respeto por los demás, competir a nivel nacional e internacional con el objetivo de ayudar a la formación del carácter y disciplina de quienes lo practican; además realiza tareas como el registro de alumnos, la creación de eventos de competencia y la elaboración de los pools o llaves de peleas de los diferentes campeonatos que realiza.

La presente investigación se basa en la elaboración de una aplicación web para la gestión administrativa de la escuela de taekwondo “Nieto’s Academia” utilizando framework React Native y Bootstrap por el lado del cliente; y entorno de ejecución NodeJs por el lado del servidor, trabajando en conjunto con la base de datos MongoDB, usando la metodología ágil “SWIRL”, siguiendo las fases de modelado, planificación, análisis, pruebas e implementación [2], para garantizar un software de calidad.

El entorno NodeJs es un framework de entrada / salida que se basa en eventos, es utilizado para crear aplicaciones que ejecutan JavaScript en el cliente como en el lado del servidor [3]; este marco de trabajo facilitará la comunicación en tiempo real a través del lenguaje JavaScript para la creación de las peticiones que necesite el sistema. Por otra parte, tenemos a React que ayuda a crear interfaces de usuario de forma sencilla para que el cliente pueda visualizar de una forma agradable una aplicación web.

La aplicación web tendrá como principal funcionalidad la automatización de los procesos que realiza la academia, es decir, la gestión administrativa de los alumnos y eventos de la misma, a su vez, se almacenará dicha información en una base de

datos permitiendo la seguridad e integridad de la información; además, contará con un módulo que permita la creación de pools o llaves de pelea de los diferentes campeonatos que se realicen.

El presente trabajo de titulación consta de 3 capítulos, los cuáles se detallan a continuación:

**Capítulo I:** Abarca la problemática del proyecto, la propuesta para la solución del mismo y la obtención de los requisitos requeridos por parte del usuario, además, se establecen las herramientas necesarias para el desarrollo del sistema.

**Capítulo II:** Hace énfasis en la elaboración de la aplicación con su ingeniería utilizando como guía la metodología de desarrollo web SWIRL.

**Capítulo III:** Finalmente se realizará el análisis de los resultados que se obtendrán en base al estándar internacional de calidad ISO/IEC 9126, determinando así la calidad del sistema.

## ÍNDICE DE CONTENIDO

<b>DEDICATORIA</b>	<b>1</b>
<b>AGRADECIMIENTO</b>	<b>2</b>
<b>RESUMEN</b>	<b>3</b>
<b>ABSTRACT</b>	<b>5</b>
<b>INTRODUCCIÓN</b>	<b>7</b>
<b>1. CAPÍTULO I. DIAGNÓSTICO DE NECESIDADES Y REQUERIMIENTOS</b>	<b>16</b>
1.1. Ámbito de Aplicación: descripción del contexto y hechos de interés	16
1.2. Establecimiento de requerimientos	17
1.3. Justificación de requerimiento a satisfacer	18
<b>2. CAPÍTULO II. DESARROLLO DEL PROTOTIPO</b>	<b>19</b>
2.1. Definición del prototipo tecnológico	19
2.1.1. Arquitectura Cliente - Servidor	19
2.1.2. Arquitectura a implementar	20
2.1.3. Arquitectura de software	21
2.1.4. Capa de presentación	22
2.1.5. Capa lógica	23
2.1.6. Capa de datos	23
2.2. Fundamentación teórica del prototipo	23
2.2.1. Aplicación web	24
2.2.2. Metodología de desarrollo de software	25
2.2.2.1. Metodología SWIRL	25
2.2.3. Cliente web	26
2.2.3.1. Framework web	27
2.2.3.2. Framework React	27
2.2.3.3. Html	27
2.2.3.4. CSS	27
2.2.3.5. Bootstrap	28
2.2.3.6. JavaScript	28
2.2.4. Servidor web	28
2.2.4.1. Framework NodeJs	28
2.2.5. Base de Datos	29
2.2.5.1. Base de datos relacionales	29
2.2.5.2. Base de datos no relacionales	29
2.2.5.2.1. MongoDB	30
2.2.6. Herramientas CASE	30
2.2.6.1. Microsoft Visio	31
2.2.6.2. Microsoft Project	31
2.2.6.3. Toad Data Modeler	31
2.2.6.4. StarUML	31
2.2.6.5. Bizagi Modeler	31

2.2.6.6. Balzamiq Mockups	32
2.3. Objetivos del prototipo	32
2.3.1. Objetivo General	32
2.3.1. Objetivos Específicos	32
2.4. Diseño del prototipo	33
2.4.1. Fase de análisis	33
2.4.1.1. Alcance del proyecto	33
Tabla 1 Funcionalidades del sistema	33
Tabla 2 Limitaciones del sistema	34
2.4.1.2. Análisis de la factibilidad	34
2.4.1.2.1. Factibilidad operativa	34
Tabla 3 Factibilidad operativa de la aplicación	34
2.4.1.2.2. Factibilidad técnica	35
Tabla 4 Factibilidad Técnica	35
2.4.1.2.3. Factibilidad económica	36
Tabla 5 Factibilidad Económica	36
2.4.1.3. Identificación de interesados o “stakeholders”	36
Tabla 6 Matriz de Interesados o Stakeholders	37
2.4.1.4. Requerimientos	37
2.4.1.4.1. Requerimientos funcionales	37
Tabla 7 Requerimientos Funcionales	38
2.4.1.4.2. Requerimientos no funcionales	38
Tabla 8 Requerimientos no funcionales	38
2.4.2. Fase de planificación	39
2.4.2.1. Historias de usuarios	39
Tabla 9 Historia de usuario N°1 - Registro al sistema	39
Tabla 10 Historia de usuario N°2 - Ingreso al sistema	40
Tabla 11 Historia N°3 - Gestión de alumnos	40
Tabla 12 Historia de usuario N°4 - Gestión de eventos	41
Tabla 13 Historia de usuario N°5 - Creación de llaves	42
Tabla 14 Historia de usuario N°6 - Gestión de usuarios	43
2.4.2.2. Gestión del cronograma	43
2.4.2.3. Gestión de riesgos	45
2.4.3. Fase de modelado	45
2.4.3.1. Diseño de base de datos no relacional	45
2.4.3.2. Diseño de interfaces	47
2.4.3.3. Diseño de diagramas UML	53
2.4.3.3.1. Diagramas de caso de uso	53
2.4.3.3.2. Diagramas de actividades	56
2.4.3.3.3. Diagramas de secuencias	59
2.5. Ejecución y/o ensamble del prototipo	61
<b>3. CAPÍTULO III. EVALUACIÓN DEL PROTOTIPO</b>	<b>69</b>
3.1. Plan de evaluación	69
3.2. Resultados de la evaluación	69

3.2.1. Evaluación de la calidad de la aplicación web	69
Tabla 15 Evaluación de calidad ISO/IEC 9126	69
3.2.1.1. Website Auditor	70
3.2.1.2. Website Grader	71
3.2.1.3. GTmetrix	72
Tabla 16 Evaluación calidad - usabilidad - portabilidad	74
3.3. Conclusiones	75
3.4. Recomendaciones	76
<b>4. BIBLIOGRAFÍA</b>	<b>77</b>
<b>5. ANEXOS</b>	<b>80</b>
ANEXO A: ANÁLISIS DE RIESGOS	80
Tabla 17 Gestión de Riesgos	80
ANEXO B: FICHA DE INSCRIPCIÓN DE LA ESCUELA “NIETO’S ACADEMIA	81

## ÍNDICE DE TABLAS

<b>Tabla 1</b> Funcionalidades del sistema.	33
<b>Tabla 2</b> Limitaciones del sistema.	34
<b>Tabla 3</b> Factibilidad operativa de la aplicación.	34
<b>Tabla 4</b> Factibilidad Técnica.	35
<b>Tabla 5</b> Factibilidad Económica.	36
<b>Tabla 6</b> Matriz de Interesados o Stakeholders.	37
<b>Tabla 7</b> Requerimientos Funcionales.	38
<b>Tabla 8</b> Requerimientos no funcionales.	38
<b>Tabla 9</b> Historia de usuario N°1 - Registro al sistema.	39
<b>Tabla 10</b> Historia de usuario N°2 - Ingreso al sistema.	40
<b>Tabla 11</b> Historia N°3 - Gestión de alumnos.	40
<b>Tabla 12</b> Historia de usuario N°4 - Gestión de eventos.	41
<b>Tabla 13</b> Historia de usuario N°5 - Creación de llaves.	42
<b>Tabla 14</b> Historia de usuario N°6 - Gestión de usuarios.	43
<b>Tabla 15</b> Evaluación de calidad ISO/IEC 9126.	69
<b>Tabla 16</b> Evaluación calidad - usabilidad - portabilidad.	74
<b>Tabla 17</b> Gestión de Riesgos.	80

## ÍNDICE DE IMÁGENES

<b>Ilustración 1</b>	Arquitectura general de la aplicación web Nieto'sApp.	21
<b>Ilustración 2</b>	Modelo de tres capas.	22
<b>Ilustración 3</b>	Fundamentación Teórica.	24
<b>Ilustración 4</b>	Ciclo de vida de la metodología SWIRL.	26
<b>Ilustración 5</b>	Cronograma de actividades.	44
<b>Ilustración 6</b>	Modelo de base de datos NoSQL Módulo Usuarios.	46
<b>Ilustración 7</b>	Modelo de base de datos NoSQL Módulo Alumnos.	46
<b>Ilustración 8</b>	Modelo de base de datos NoSQL Módulo Eventos.	47
<b>Ilustración 9</b>	Diseño de Login-Registro.	47
<b>Ilustración 10</b>	Diseño de la Página de Inicio.	48
<b>Ilustración 11</b>	Diseño de la Página de Alumnos.	49
<b>Ilustración 12</b>	Diseño de la creación de un alumno.	49
<b>Ilustración 13</b>	Diseño del buscador de alumnos.	50
<b>Ilustración 14</b>	Diseño de la Página de Eventos.	51
<b>Ilustración 15</b>	Diseño de la creación de un evento.	51
<b>Ilustración 16</b>	Diseño de la creación de pools de pelea.	52
<b>Ilustración 17</b>	Diagrama de caso de uso - General	53
<b>Ilustración 18</b>	Diagrama de caso de uso - Iniciar Sesión.	54
<b>Ilustración 19</b>	Diagrama de caso de uso - Registrar Usuario.	54
<b>Ilustración 20</b>	Diagrama de caso de uso - Administración de Alumnos.	55
<b>Ilustración 21</b>	Diagrama de caso de uso - Administración de Eventos.	55
<b>Ilustración 22</b>	Diagrama de caso de uso - Creación de pools de pelea.	56
<b>Ilustración 23</b>	Diagrama de actividades - Inicio de Sesión.	56
<b>Ilustración 24</b>	Diagrama de actividades - Registro de Usuario.	57
<b>Ilustración 25</b>	Diagrama de actividades - Administración de Alumnos.	57
<b>Ilustración 26</b>	Diagrama de actividades - Administración de Eventos.	58
<b>Ilustración 27</b>	Diagrama de Actividades - Creación de llaves de competencia.	58

<b>Ilustración 28</b>	Diagrama de secuencia - Inicio de Sesión.	59
<b>Ilustración 29</b>	Diagrama de secuencia - Administrar Alumnos.	59
<b>Ilustración 30</b>	Diagrama de secuencia - Administrar Eventos.	60
<b>Ilustración 31</b>	Diagrama de secuencia - Crear llaves de pelea.	60
<b>Ilustración 32</b>	Implementación - Página Login-Registro.	61
<b>Ilustración 33</b>	Implementación - Registro.	62
<b>Ilustración 34</b>	Implementación - Iniciar Sesión.	62
<b>Ilustración 35</b>	Página Principal	63
<b>Ilustración 36</b>	Página Alumnos.	64
<b>Ilustración 37</b>	Formulario para crear un alumno.	64
<b>Ilustración 38</b>	Información del Alumno.	65
<b>Ilustración 39</b>	Modificar Alumno.	65
<b>Ilustración 40</b>	Eliminar un Alumno.	66
<b>Ilustración 41</b>	Página Buscar Alumno.	66
<b>Ilustración 42</b>	Búsqueda de un alumno.	67
<b>Ilustración 43</b>	Página de Eventos.	68
<b>Ilustración 44</b>	Formulario para la creación de un evento.	68
<b>Ilustración 45</b>	Evaluación calidad - funcionalidad del sistema.	70
<b>Ilustración 46</b>	Evaluación calidad - funcionalidad del sistema.	70
<b>Ilustración 47</b>	Evaluación calidad – funcionalidad del sistema.	71
<b>Ilustración 48</b>	Evaluación de calidad - eficiencia y mantenibilidad.	72
<b>Ilustración 49</b>	Evaluación de seguridad - GTmetrix.	73
<b>Ilustración 50</b>	Ficha de inscripción "Nieto's Academia"	81

## **1. CAPÍTULO I. DIAGNÓSTICO DE NECESIDADES Y REQUERIMIENTOS**

### **1.1. Ámbito de Aplicación: descripción del contexto y hechos de interés**

Taekwondo es una de las artes marciales tradicionales más sistemáticas y científicas de Corea [4] que ayuda a fortalecer el espíritu y la vida a través del entrenamiento del cuerpo y la mente, actualmente es considerada una de las más prácticas en el mundo razón por la cual se ha convertido en un deporte global y forma parte de los Juegos Olímpicos; su importancia radica en su utilidad como herramienta didáctica con la finalidad de desarrollar las capacidades físicas y cognitivas en los niños, permitiendo mejorar aspectos como: psicomotriz, conductual y salud [5].

La historia del Taekwondo en el Ecuador se remonta al año de 1969 con el arribo de dos maestros coreanos, en Guayaquil, el Maestro Bum Jae Lee quién expandió el estilo Jidokwan entre sus alumnos, mientras que en Quito el Maestro Chul Woong Jang enseñó el estilo Chundokwan [4], poco a poco el taekwondo se fue dando a conocer en las demás provincias del país hasta convertirse en un deporte nacional; en la provincia de El Oro esta arte marcial se inició en el año de 1984 con la llegada del profesor Lic. Melchor León, pionero que introdujo este deporte en la provincia; los hechos relatan que un 3 de febrero de ese año, se dictó la primera clase de la mano de León, dirigida a novatos y aficionados, con 50 alumnos y una duración de dos meses.

En la historia del Taekwondo Orense han resaltado grandes deportistas que han dejado en alto el nombre de la provincia y del país; atletas como las hermanas Franco, Richard Borbor, Gerardo Nieto, Adrián Miranda, entre otros, son las estrellas con las que ha contado la provincia; un atleta a destacar y que forma parte de la historia es Erick Fabricio Arévalo Miranda, quién fue el primer Machaleño en llegar a unos Juegos Olímpicos de la Juventud realizados en Nanjing China en el año 2014 adjudicándose el 5to lugar en dicho certamen.

En la actualidad el Taekwondo es uno de los deportes más practicados en el país, fomentado no sólo por parte de la Federaciones provinciales sino también por escuelas privadas que enseñan esta arte marcial con el objetivo de aportar en la salud y bienestar de los más pequeños; en la ciudad de Machala existen alrededor

de unas 7 escuelas que instruyen esta actividad, sin embargo a nivel provincial y nacional este número incrementa significativamente, impulsando de esta manera la actividad deportiva en niños, niñas y jóvenes.

Nieto's Academia es una escuela de Taekwondo que instruye este arte marcial en la ciudad de Machala, su principal enfoque es el entrenamiento deportivo, de niños, niñas y jóvenes en general, donde se puede aprender defensa personal, aumentar la condición física, la formación del carácter y sobre todo mejorar la disciplina; esta entidad realiza procesos como el seguimiento de sus alumnos y la creación de eventos de competencia a nivel nacional careciendo de algún tipo de tecnología que gestione y automatice dichos tareas; por lo cual, dichas labores son realizadas de manera manual, resultando un trabajo tedioso y poco efectivo.

Como proyecto de titulación se busca satisfacer la problemática mencionada anteriormente a través de la incorporación de la tecnología mediante el desarrollo de una aplicación web utilizando herramientas como React y NodeJs para la automatización de los procesos que realiza la academia como el registro de alumnos, la creación de eventos y la elaboración de pools o llaves de competencia

## **1.2. Establecimiento de requerimientos**

Nieto's Academia Taekwondo realiza actividades como el registro de los alumnos y la realización de eventos o campeonatos, dentro de los eventos se realizan los respectivos pools o llaves de peleas de las competencias que se llevarán a cabo; estas tareas se las realiza de forma manual, por lo cual, no se mantiene un debido control de cada una de ellas.

El presente trabajo de titulación pretende enfocarse en la optimización y automatización de cada uno de los procesos mencionados anteriormente permitiendo lograr una mejor adaptabilidad, portabilidad y eficiencia, y la seguridad de los datos que posee la academia mediante la implementación de una aplicación web; la ventaja de utilizar esta tecnología es que se tendrá la información en un solo lugar garantizando la comodidad de sus propietarios.

Para el desarrollo de este sistema se utilizó el proceso de la ingeniería de requerimientos con los interesados del proyecto [6], es importante destacar que utilizar esta técnica ayuda a identificar el conjunto de actividades y tareas que se requiere realizar en la aplicación, lo que permite elaborar un sistema estable,

eficiente y de calidad; una vez establecidos los requisitos principales se obtuvo la siguiente esquematización de los mismos:

- Gestión de perfiles de usuarios. Un perfil administrador y usuarios registrados; de este modo se contará con un único usuario que tendrá acceso total a todas las funciones del sistema (administrador) mientras que los demás tendrán acceso limitado a la aplicación.
- Gestión de alumnos. Creación, edición y eliminación de los mismos.
- Gestión de eventos de competencia.
- Generación automática de pools o llaves de competencia.

Por otra parte, los requisitos funcionales y no funcionales serán especificados y detallados en la fase de análisis de la metodología a emplear.

### **1.3. Justificación de requerimiento a satisfacer**

El auge tecnológico se encuentra presente en diversas áreas como la educación, la salud, la industria, las telecomunicaciones, la empresarial, entre otras, y no es una excepción utilizarla dentro del ámbito deportivo; por medio del uso de tecnologías de información muchas empresas han logrado mejorar sus servicios, catapultarlos y optimizar cada uno de sus procesos; industrias como la deportiva utilizan los medios tecnológicos para mejorar sus técnicas de entrenamiento y conseguir mejores resultados en los deportistas.

El presente proyecto tiene como finalidad mejorar notablemente los procesos realizados por la escuela de Taekwondo “Nieto’s Academia” mediante una aplicación web que gestione las tareas administrativas de la entidad como el registro de los alumnos, la realización de eventos de competencia y la creación de los pools o llaves de peleas que son necesarios para el desarrollo de los campeonatos.

Debido a la gran cantidad de datos que maneja la academia, es necesario la implementación de un sistema dinámico, rápido, seguro, estable y que funcione con una arquitectura “cliente - servidor” [7], a través de una infraestructura que sea capaz de soportar constantes solicitudes de información por parte de diferentes usuarios al mismo tiempo permitiendo utilizar todos los servicios que ofrece la aplicación.

En cuanto a los roles y permisos, se maneja dos perfiles o niveles de usuario, los cuales son: (Perfil 1), administrador, el cual se encargará del manejo total de la

aplicación teniendo acceso a todas las funciones que realice el sistema; (Perfil 2), corresponde a los usuarios que se registren en la aplicación, los mismos que solo tendrán acceso limitado a la plataforma, esto permite que la aplicación web sea totalmente administrable.

Todos los requerimientos planteados serán abarcados por medio de la aplicación de las tendencias tecnológicas en el desarrollo web tanto por el lado del “Back-end” (servidor) como en el “Front-end” (cliente) utilizando la metodología ágil “SWIRL” haciendo énfasis en sus fases muy bien definidas que permitan garantizar la elaboración de un software de calidad mediante la planificación, desarrollo y ejecución de las pruebas de rendimiento del sistema.

## **2. CAPÍTULO II. DESARROLLO DEL PROTOTIPO**

### **2.1. Definición del prototipo tecnológico**

El presente proyecto se encuentra estructurado de dos partes principales que son el “Back-end” para la parte del servidor y el “Front-end” para la parte visual, es decir, la que manipulará el usuario; en este sentido se utilizarán herramientas que actualmente son tendencia en el desarrollo web, estas tecnologías permitirán tener una mejor visualización y distribución de información entre diferentes servidores. A continuación, se describe en detalle cómo se distribuye internamente el sistema.

#### **2.1.1. Arquitectura Cliente - Servidor**

La arquitectura cliente – servidor es uno de los modelos de aplicación distribuidos más conocidos en el que las transacciones se dividen en procesos que cooperan entre sí para el intercambio de información; consta de dos componentes, el proveedor o servidor que brinda una serie de servicios que son solicitados y extenuados por el cliente o consumidor [8].

La principal importancia de este modelo es permitir que varios clientes se conecten al mismo tiempo a los servicios proporcionados por el servidor, pues, actualmente la mayoría de aplicaciones debido a su gran demanda deben ser utilizados por múltiples usuarios simultáneamente.

- Entre las ventajas de esta arquitectura se tienen las siguientes:
- Mejora el servicio destinado a los clientes.

- Acrecienta la productividad, mejora el intercambio y la consistencia de la información, reduce los costos operativos.
- Disminuye el tráfico en la red.
- Tiempo de desarrollo reducido permitiendo la reutilización, portabilidad, modularidad.

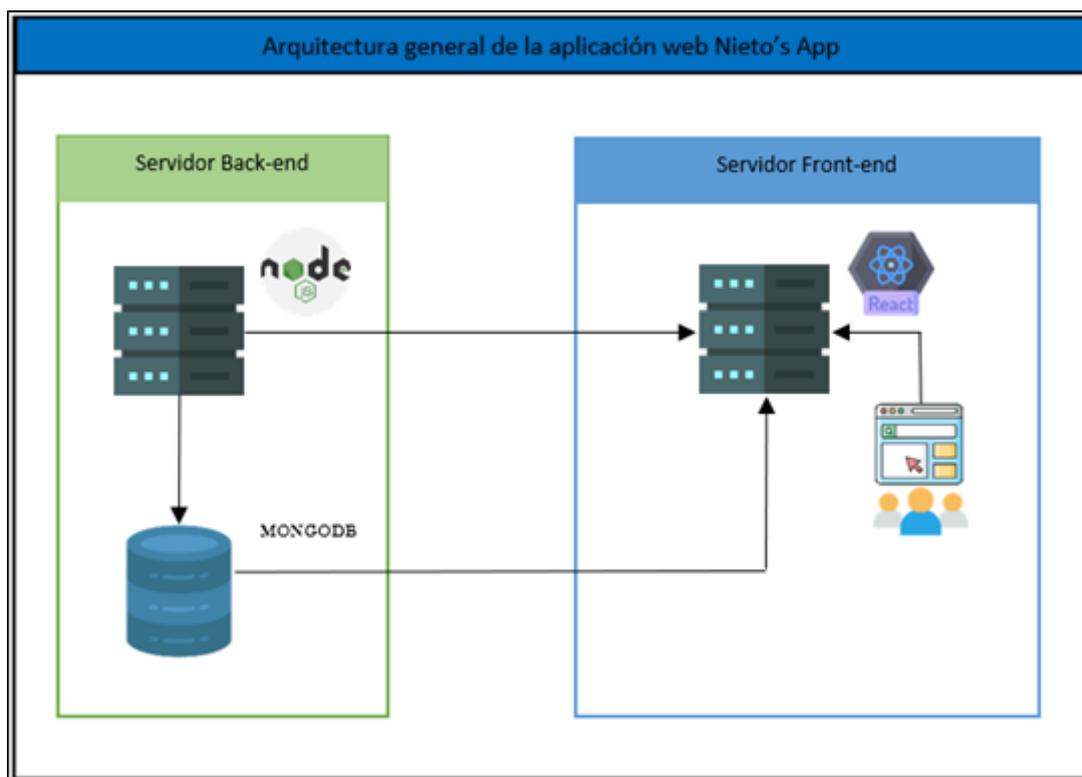
### **2.1.2. Arquitectura a implementar**

La aplicación web a desarrollar requiere de mantener comunicaciones de información entre diferentes equipos computacionales y clientes conectados al mismo tiempo, por ese motivo se implementó la arquitectura cliente – servidor, ya que, es la tecnología que brinda a los usuarios finales acceso transparente a aplicaciones, datos o servicios mediante la organización en múltiples plataformas [9]; este sistema se divide en dos partes principales denominadas “Front-end” y “Back-end”.

El “Back-end” es la parte de la aplicación que los usuarios finales no pueden ver, su función es acceder a la información solicitada a través del sistema para luego combinarla y devolverla al cliente final; de esta manera se cuenta con un servidor encargado de recibir todas las peticiones, el cual, está compuesto por una API que contiene todas las solicitudes que el operador necesite.

El “Front-end” es la capa superficial de la aplicación que contiene imágenes, botones, menús, entre otros, es decir, es la interfaz gráfica que el usuario puede observar y manipular, tiene una relación directa con el “Back-end”, pues, es el encargado de darle funcionamiento a todo lo que contiene el “Front-end”; de esta forma se tiene un servidor con los frameworks necesarios para que el usuario puede visualizar la información. A continuación, se muestra la distribución de los servidores.

**Ilustración 1** Arquitectura general de la aplicación web Nieto's App



Fuente: Elaboración propia

Como se puede observar en la **ilustración 1**, la arquitectura se divide en dos partes principales (Back-end y Front-end), donde el servidor Back-end es el encargado de realizar todas las peticiones solicitadas por el usuario, de tal manera que podrá observar la información mediante el servidor Front-end.

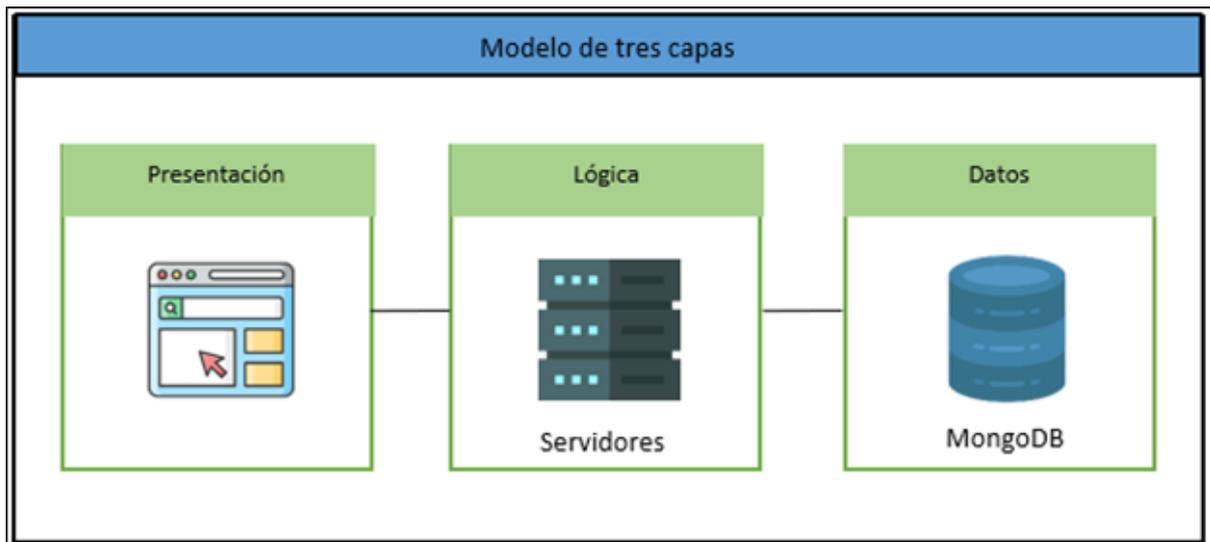
### 2.1.3. Arquitectura de software

La arquitectura de software describe la estructura de un sistema, por lo tanto, se considera como una parte temprana de la fase de diseño [10], ya que reúne los requisitos técnicos y operativos para el desarrollo de un sistema, los mismos que son difíciles de cambiar; su importancia está en satisfacer los requerimientos no funcionales, que están relacionados a los atributos de calidad como lo son la seguridad, el rendimiento y la escalabilidad [11].

La aplicación web propuesta requiere de ser distribuida por capas con la finalidad de evitar que se vea afectado todo el sistema en el caso de que exista alguna modificación a futuro, lo que permite tener una mejor flexibilidad en el desarrollo general de la plataforma; es por ello que el proyecto empleará el modelo de tres capas.

La arquitectura o modelo de tres capas es usada en la gran mayoría de sistemas incluidas las plataformas web, su función principal es dividir la aplicación en tres partes diferenciadas, lo que permite que cada capa solo se comuniquen con la anterior; la primera se encarga de la presentación al usuario final, la segunda de la lógica de datos del sistema y la última de manipular la información [12].

**Ilustración 2** Modelo de tres capas



**Fuente:** Elaboración propia

La **ilustración 2**, presenta la arquitectura de software a utilizar en el desarrollo de la aplicación web, como se puede observar se trata de un modelo de tres capas, cada una de ellas diferenciadas entre sí; la primera es la que podrá visualizar el cliente (presentación), la segunda es en donde se encuentran los servidores (lógica del sistema) y la tercera se refiere a la base de datos (manipulación de la información).

#### **2.1.4. Capa de presentación**

La capa de presentación es aquella sobre la que se construyen los diseños o plantillas que se presentan al cliente, esto se denomina "Front-end" y es en donde se lleva a cabo la interacción del usuario con la funcionalidad proporcionada por la aplicación; de este modo su función es pasarle todas las peticiones que solicite el usuario a la capa de la lógica del negocio [13].

Para el desarrollo del sistema, en lo referente al "Front-end" se utilizará el framework React debido a que ayuda a crear las diversas interfaces de usuario necesarias de una forma sencilla y dinámica aprovechando al máximo todos los recursos con los que se cuenta.

### **2.1.5. Capa lógica**

Esta capa se encarga de gestionar toda la funcionalidad de la aplicación, es decir, cubre todos los procesos involucrados en la gestión de la información; en otras palabras, actúa como intermediario entre los datos almacenados en la base de datos y el usuario que solicita esta información para realizar sus funciones con mayor eficiencia [12]; para llevar a cabo lo mencionado anteriormente existe un gran número de lenguajes de programación cuyo objetivo es facilitar este proceso como lo son Python, Java, Ruby, PHP, JavaScript, entre otros.

En cuanto a la propuesta planteada, se trabajará con el entorno de ejecución NodeJs, el cual permitirá realizar toda la lógica y funcionalidad del sistema para luego ser utilizado en la capa de presentación y tener así un sistema rápido, eficiente, seguro, confiable y de calidad.

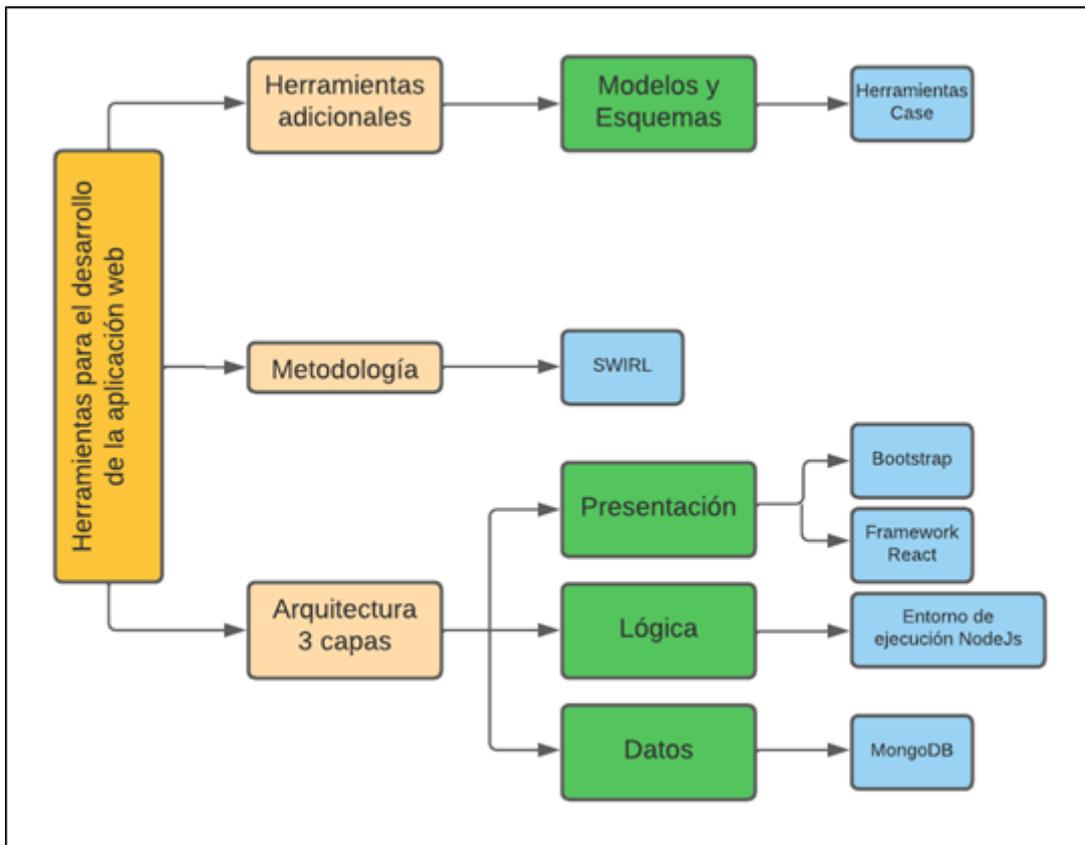
### **2.1.6. Capa de datos**

Esta capa corresponde a las tareas habituales que se realiza con los datos como insertar, modificar, consultar y eliminar, es decir, se refiere al almacenamiento de todo tipo de información ya sea de usuarios, compras, ventas, inventario, entre otros en la base de datos y los diferentes procesos que se realicen en ella [12]; para el desarrollo del sistema planteado se utilizará un sistema de gestor de bases de datos NoSQL conocido como MongoDB por su escalabilidad, flexibilidad y alta disponibilidad al momento de realizar un proyecto [14].

## **2.2. Fundamentación teórica del prototipo**

El presente proyecto tecnológico tiene una base sólida de fundamentación que permite la administración de software de calidad; en la **ilustración 3** se detalla la estructura de los componentes, herramientas y tecnologías aplicadas para la elaboración de la plataforma web, de la misma manera se especifica el marco de trabajo utilizado para lograr un sistema eficiente.

**Ilustración 3** Fundamentación Teórica



Fuente: Elaboración propia

### 2.2.1. Aplicación web

Una aplicación web es una colección de recursos que se ejecutan desde un navegador web sin necesidad de distribuir, instalar o actualizar programas adicionales para el usuario [15]; en otras palabras, son un grupo de páginas dinámicas con el objetivo principal de permitir que el usuario pueda realizar una o varias tareas simultáneamente como navegar por internet, editar una imagen de manera online, escuchar música, comprar en línea, entre otras.

Las aplicaciones web se desarrollan de acuerdo a diferentes lineamientos establecidos en los marcos de trabajo de desarrollo de software y, a diferencia de las aplicaciones móviles y de escritorio, se caracterizan por su capacidad para ejecutarse dinámicamente en cualquier plataforma [16]; actualmente son muy utilizadas en diversas áreas y para realizar variedad de tareas, se pueden encontrar en bancos y agencias gubernamentales, en redes sociales, en empresas de diferente índole e incluso en el comercio digital o e-commerce.

## **2.2.2. Metodología de desarrollo de software**

Una metodología de desarrollo de software es un marco efectivo que brinda respuestas a los problemas emergentes, enfocado en brindar mejoras significativas del producto a través de etapas o procesos efectivos que tienen como objetivo promover la calidad de una aplicación [17].

El crecimiento constante de las compañías y su conexión con las tecnologías actuales hace que el desarrollo de software se encuentre en constante cambio, por ese motivo las metodologías utilizadas para aplicaciones tradicionales no son efectivas para la elaboración de plataformas web porque pierden las características relevantes de este tipo de sistema [18].

Las metodologías ágiles son las más utilizadas por los desarrolladores porque resuelven fácilmente los problemas que surgen durante la implementación, como cambios persistentes en los requisitos del cliente, falta de seguridad, problemas de diseño y componentes del sistema de implementación [19].

Las metodologías híbridas incluyen agrupar diferentes marcos de trabajo basados en las debilidades de las metodologías tradicionales y ágiles, con el objetivo de convertirse en una metodología robusta y flexible proporcionando nuevos métodos para el desarrollo de software [20].

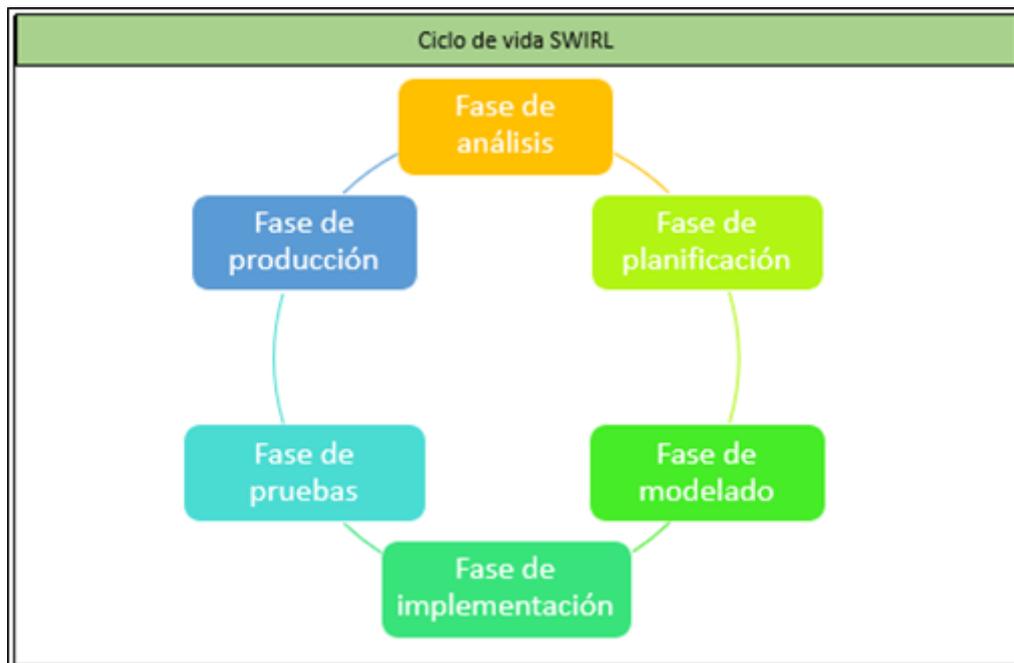
Utilizar una metodología adecuada durante un proyecto es un punto importante para lograr una mejor calidad en la producción del mismo; basándose en los trabajos de titulación realizados desde el año 2019 hasta la actualidad el repositorio de la Universidad Técnica de Machala [21] manifiesta que en las tesis referentes a propuestas tecnológicas y al desarrollo web la mayoría de estudiantes utilizan el marco de trabajo para el desarrollo de software ágil "SWIRL" gracias a sus fases muy bien definidas que permiten la elaboración de una aplicación eficiente.

### **2.2.2.1. Metodología SWIRL**

Software Web Iterativo Relacional Lógico o SWIRL, es un marco de trabajo centrado en sistemas web que une enfoques híbridos e iterativos basado en la implementación de iteraciones en cada una de sus etapas, además, permite que el usuario se encuentre presente durante todo el proceso de desarrollo y garantiza la incorporación de la etapa de modelo de negocios [2].

Esta metodología considera en cada interacción los cinco criterios del desarrollo de proyectos, que son: el costo, tiempo, calidad, alcance y comunicaciones; entre sus características principales se tiene las siguientes: reduce los costos de recursos humanos, el tiempo de desarrollo y de implementación; los requerimientos de usuario pueden modificarse en cualquier etapa del proyecto y las partes interesadas deben estar bien definidas; por otro lado, su ciclo de vida posee seis fases, las mismas que se muestran a continuación:

**Ilustración 4** Ciclo de vida de la metodología SWIRL



**Fuente:** Elaboración propia

La **ilustración 4**, presenta las fases del ciclo de vida de la metodología SWIRL, las mismas que forman parte de un proceso iterativo, desde el análisis de los requisitos hasta la puesta en marcha en la fase de producción.

### **2.2.3. Cliente web**

Un cliente web, es cualquier aplicación que pueda acceder a un servidor y solicitar servicios o información específica; la tarea del cliente es estandarizar las consultas, transferirlas al servidor y preparar los datos resultantes para poder ser visualizados [22]; para su desarrollo utiliza tecnologías “Front-end” basadas en lenguaje html, CSS y JavaScript.

### **2.2.3.1. Framework web**

Un framework web es una colección de técnicas y conocimientos cuya adaptación permite el uso de materiales y medios naturales a través de la invención y fabricación [23]; por otro lado, un marco web es una estructura predefinida cuyos componentes se pueden reutilizar, lo que facilita la elaboración de aplicaciones en tiempos reducidos y poco mantenimiento [24].

En el desarrollo web es necesario el uso de estos frameworks debido a que incluyen librerías, soportes, lenguajes, entre otras herramientas, lo que facilita la elaboración de un sistema web.

### **2.2.3.2. Framework React**

React es una biblioteca de JavaScript que permite crear interfaces de usuario interactivas con facilidad [25]; las aplicaciones construidas con React están escritas en un estilo orientado a objetos que consisten en componentes organizados en una estructura de árbol; cada componente consta con un conjunto de propiedades, un estado con valores que pueden cambiar con el tiempo y métodos que especifican los subcomponentes del componente principal [26]; estas características lo convierten en el framework web más utilizado en la actualidad.

### **2.2.3.3. Html**

Html (Lenguaje de Marcado de Hipertexto) es el código con el que se escriben, estructuran y despliegan las páginas web; consiste en una variedad de elementos que permiten envolver diferentes contenidos para que se vean o comporten de diferente manera [27]; en otras palabras, es el lenguaje utilizado por los navegadores para que el usuario pueda observar las aplicaciones web.

### **2.2.3.4. CSS**

CSS es una tecnología que se utiliza para definir el estilo o la apariencia de las páginas web escritas con documentos Html o Xml; está diseñado para separar el contenido, la estructura y la parte visual de la web brindando a los desarrolladores más control sobre el aspecto del sistema que estén realizando [28].

### **2.2.3.5. Bootstrap**

Bootstrap es un conjunto de herramientas “Front-end” potente y rico en funciones que permite diseñar sitios web desde un prototipo hasta la producción [29]; en otras palabras, es un framework de CSS con el que se puede elaborar aplicaciones web responsivas y visualmente usables para el usuario; su ventaja es que reduce la creación de archivos CSS comúnmente utilizados [30].

### **2.2.3.6. JavaScript**

JavaScript es un lenguaje orientado a objetos ligero, interpretado y funcional de primera clase, más conocido como lenguaje de secuencias de comandos para páginas web, además, es utilizado en entornos como NodeJs, Apache CouchDB y Adobe Acrobat [31]; actualmente es el lenguaje de programación más utilizado para el desarrollo web junto con Html y CSS, esta combinación hace posible la funcionalidad de la mayoría de aplicaciones web existentes [32].

## **2.2.4. Servidor web**

Un servidor web es una aplicación que responde a las solicitudes de los navegadores web y proporciona los recursos solicitados a través del protocolo HTTP o de forma segura a través de HTTPS [33]; en otras palabras, es un software que forma parte de un servidor y tiene como objetivo principal devolver la información cuando recibe peticiones realizadas por parte de los usuarios [34].

### **2.2.4.1. Framework NodeJs**

Es una plataforma de ejecución de JavaScript orientado a eventos asíncronos que permite crear aplicaciones web escalables y, en combinación con ExpressJs, un framework que se desempeña con NodeJs, facilita la generación de rutas para el protocolo HTTP [35]; su principal característica es que usa JavaScript en el servidor y a su vez es multiplataforma lo que lo convierte en uno de los frameworks más potentes y utilizados del mercado.

Utiliza una arquitectura modular basada en eventos y su estilo de programación asíncrona permiten que NodeJs funcione de manera estable y eficiente bajo cargas de trabajo pesadas [36]; de esta manera permite realizar una gran cantidad de tareas del lado del servidor como generar contenido dinámico, recopilar datos de

formulario y realizar operaciones crud tanto en la base de datos como en el propio servidor.

### **2.2.5. Base de Datos**

Una base de datos es una colección de información estructurada presentada de manera accesible y actualizada, se clasifica según el tipo de contenido que puede ser: bibliográfico, textual, numérico y pictórico [37]; su principal función es mantener los datos almacenados, íntegros y seguros permitiendo acceder a ellos en cualquier momento que se necesite.

Por otro lado tenemos a los sistemas gestores de bases de datos, los mismos que presentan las siguientes características: crean independencia de datos, garantizan la integridad de los mismos, aseguran la estabilidad, se recuperan de errores y proporcionan la seguridad necesaria de la información [14]; existen dos sistemas de bases de datos, los estructurados llamados relacionales y los no estructurados o no relacionales.

#### **2.2.5.1. Base de datos relacionales**

Las bases de datos relacionales son sistemas que manejan el lenguaje de consulta estructurada SQL, utilizando tablas, es decir, el modelo de filas y columnas, en donde la información es almacenada y procesada basándose en el diseño relacional realizado [38].

#### **2.2.5.2. Base de datos no relacionales**

Las bases de datos no relacionales o NoSQL constan con una estructura dinámica para el almacenamiento de la información, donde la velocidad y el volumen de datos es lo principal, creando sistemas más distribuidos [39]; su característica principal radica en su facilidad de desarrollo, tanto en términos de funcionalidad como en rendimiento, utiliza diferentes modelos de datos como documentos, gráficos, entre otros, lo que las convierte en una poderosa administradora de información. Entre las características de este tipo de bases de datos tenemos las siguientes:

- Mayor escalabilidad horizontal y vertical.
- Soporte de estructura distribuida.
- Son más flexibles.

- Permiten realizar cambios en los esquemas sin necesidad de detener la Base de datos.
- Pueden manejar grandes cantidades de datos de manera óptima.

#### **2.2.5.2.1. MongoDB**

MongoDB es la más popular entre las bases de datos tipo NoSQL, es una gran herramienta para crear un almacén de datos que ofrece alto rendimiento, alta disponibilidad, flexibilidad y automatización escalada; además, permite manejar colecciones que contienen diferentes documentos como archivos JSON, BSON y ficheros de más de 16 MB de almacenamiento como por ejemplo vídeos [40].

El segundo aspecto a considerar es que brinda a los desarrolladores las funciones necesarias para cumplir con los requerimientos más complejos a cualquier escala [41]; a diferencia de las bases de datos relacionales que utilizan tablas, Mongo utiliza archivos planos en formato JSON [42] para almacenar la información, este tipo de registros son muy utilizados en la actualidad por un gran número de aplicaciones por lo cual la integración entre Mongo y estos sistemas es mucho más sencillo. Entre las ventajas de este gestor de bases de datos se tienen las siguientes:

- Ideal para entornos con pocos recursos computacionales.
- Herramienta con un coste relativamente bajo.
- Posee una gran documentación.
- Es perfectamente compatible con JavaScript lo que permite utilizar toda la potencia de sus funciones.

#### **2.2.6. Herramientas CASE**

Son una variedad de aplicaciones informáticas incluidas las herramientas de diseño y creación de interfaz de usuario que se utilizan para acelerar las actividades en el proceso de desarrollo de software y mejorar la comprensión del grupo de trabajo [43]; a su vez, permiten aumentar la productividad reduciendo sus costos en cuanto a tiempo y dinero. Estas herramientas ayudan en todos los aspectos del ciclo de vida del desarrollo del sistema en tareas como bosquejo de procesos, cálculo de costes, planificación, entre otras.

A continuación, se describirán algunas de las herramientas CASE que serán de gran utilidad para la elaboración del sistema web.

#### **2.2.6.1. Microsoft Visio**

Es una herramienta ideal para escalar soluciones personalizadas basadas en diagramas, topología de requisitos y desarrollo de casos comerciales, permite colaborar en tiempo real para cumplir con las especificaciones de diseño del sistema [44]; esta aplicación ayuda a organizar visualmente las ideas complejas, usar cientos de plantillas dependiendo de lo que se requiera realizar, crear diagramas dinámicos y muchas funciones más.

#### **2.2.6.2. Microsoft Project**

Herramienta con la cual se puede crear planes de actividades y diagramas de “Gantt” para simplificar la administración de sistemas complejos a través de una interfaz simple y dinámica, permitiendo enfocar o dirigir procesos de gestión de proyectos de usos múltiples [45]; en otras palabras, esta aplicación ayuda a la creación del cronograma de actividades necesarias para el desarrollo de un proyecto garantizando el cumplimiento del mismo en los tiempos establecidos.

#### **2.2.6.3. Toad Data Modeler**

Toad data Modeler crea sentencias DDL a partir de modelos lógicos y físicos, define claves primarias y foráneas y las relaciones entre ellas para que pueda modelar diferentes bancos de datos independientemente de la estructura SQL que manejen, facilitando la creación de las tablas en los SGBD o sistemas gestores de bases de datos [46].

#### **2.2.6.4. StarUML**

StarUML es una aplicación para el modelado de software ágil y sencilla, permite la creación de modelos UML y todo tipo de esquemas para el desarrollo de un sistema como diagramas casos de uso, de secuencias, de flujo, entre otros; con la capacidad de generar código mediante programación a través del modelo de clases [47].

#### **2.2.6.5. Bizagi Modeler**

Bizagi es una herramienta intuitiva y colaborativa de mapeo de procesos de negocio y actividades para optimizar los flujos de trabajo, ejecutar simulaciones y aumentar la eficiencia de los procesos dentro de una organización [48].

#### **2.2.6.6. Balzamiq Mockups**

Es una aplicación para la creación de interfaces de manera previa al desarrollo que representan visualmente los posibles resultados de la aplicación web, enriqueciendo interactivamente los requisitos funcionales del sistema que se está implementando [49].

### **2.3. Objetivos del prototipo**

#### **2.3.1. Objetivo General**

Desarrollar una aplicación web mediante el uso de framework React Native y entorno de ejecución NodeJs para la gestión administrativa de la escuela de taekwondo “Nieto 's Academia”.

#### **2.3.1. Objetivos Específicos**

- Identificar los requisitos funcionales y no funcionales a partir de las historias de usuarios.
- Utilizar el framework React Native para la elaboración del Front-end de la aplicación web.
- Implementar el entorno de ejecución NodeJs para el desarrollo del Back-end de la aplicación web.
- Implementar la metodología SWIRL mediante la aplicación de sus fases de desarrollo.
- Evaluar la calidad de la aplicación web mediante el estándar ISO/IEC 9126.

## 2.4. Diseño del prototipo

El bosquejo de la aplicación web se encuentra fundamentado en la metodología SWIRL, por lo tanto, su elaboración se la realiza basándose en las fases de desarrollo de la misma que son análisis, planificación, modelado, implementación, revisión y pruebas, las cuales serán detalladas a continuación:

### 2.4.1. Fase de análisis

La fase de análisis al ser la primera de la metodología “SWIRL” se enfoca en determinar las metas, objetivos, los requisitos funcionales y no funcionales, el alcance, los costos y las limitaciones de la aplicación a desarrollar [2]; es importante considerar que esta fase se la debe realizar previo al diseño y desarrollo del sistema para garantizar que se cumpla con lo solicitado por el cliente.

#### 2.4.1.1. Alcance del proyecto

El alcance del proyecto se deriva de un análisis de las capacidades del sistema, para cumplir con los objetivos previamente planteados.

**Tabla 1** Funcionalidades del sistema

FUNCIONALIDADES DE NIETO'S APP
Gestión de Usuarios
Gestión de Alumnos
Gestión de Eventos
Generación de llaves de competencia

**Fuente:** Elaboración propia

Basándose en las funcionalidades del sistema que se pueden observar en la **Tabla 1** se establecen las limitaciones que tendrá en el desarrollo del mismo.

**Tabla 2** Limitaciones del sistema

LIMITACIONES DE NIETO'S APP	
Usuarios no admin tendrán acceso netamente visual, no podrán manipular la información.	
Las llaves de competencia solo se crearán en los eventos que realice la academia.	

**Fuente:** Elaboración propia

Como se puede observar en la **tabla 2**, las limitaciones del proyecto identificadas son aquellos procesos que no tienen prioridad en el desarrollo del mismo.

#### **2.4.1.2. Análisis de la factibilidad**

El análisis de factibilidad se refiere a la evaluación de los recursos necesarios que se posee para el desarrollo del sistema y la estimación de los resultados, para verificar si la implementación de los mismos conducirá a pérdidas o ganancias [2].

##### **2.4.1.2.1. Factibilidad operativa**

La parte operativa del sistema estará acorde a las necesidades de los clientes y contará con personal que utilizará constantemente el sistema para las operaciones que realiza la academia. En la **tabla 3** se presenta el respectivo análisis de la factibilidad operativa para el desarrollo del sistema.

**Tabla 3** Factibilidad operativa de la aplicación

FACTIBILIDAD OPERATIVA		
NÚMERO	ACTIVIDAD	PRIORIDAD
1	Comprar un dominio para la aplicación	Alta
2	Disponibilidad de equipos de cómputo para ejecutar la aplicación	Alta

3	Conocimientos avanzados de computación en los usuarios	Baja
4	Capacitaciones sobre el manejo la aplicación	Medio

Fuente: Elaboración propia

#### 2.4.1.2.2. Factibilidad técnica

La factibilidad técnica es una evaluación del sistema en función de su puesta en marcha y desarrollo que permite conocer los recursos necesarios para su funcionamiento; en la **tabla 4** presentada a continuación se especifica los medios hardware y software.

Tabla 4 Factibilidad Técnica

FACTIBILIDAD TÉCNICA			
Tipo de recurso	Recurso	Descripción	Cantidad
HARDWARE	Procesador	Intel Core i5-10600k 4.10GHz	1
	Memoria RAM	DDR4 16GB	2
	Disco Duro	Nvme 1TB	1
SOFTWARE	MongoDB	Sistema Gestor de Base de Datos	1
	Front-end	React Native	1
	Back-end	NodeJs	1

Fuente: Elaboración propia

En la tabla 4, se puede observar las características necesarias para el desarrollo de la aplicación web tanto para los recursos de hardware como de software; contar con buenos medios permitirá garantizar la calidad respectiva.

#### 2.4.1.2.3. Factibilidad económica

La factibilidad económica permite evaluar los costos para el desarrollo de la aplicación web, debido a que el proyecto es una propuesta de trabajo de titulación no se toman en cuenta los gastos de codificación y pruebas de rendimiento; a continuación, se detalla los costos de cada tipo de recurso.

**Tabla 5** Factibilidad Económica

FACTIBILIDAD ECONÓMICA			
Tipo de recurso	Cantidad	Descripción	Costo
HARDWARE	1	PC Intel i5 10-600k / 32 GB RAM / 1 TB Nvme almacenamiento	\$900
	TOTAL		\$900
SOFTWARE	1	MongoDB	\$0
	1	Visual Studio Code	\$0
	TOTAL		\$0

Fuente: Elaboración propia

#### 2.4.1.3. Identificación de interesados o “stakeholders”

La gestión de los interesados es un paso fundamental en el desarrollo de aplicaciones web. Los stakeholders son aquellas personas que tienen interés en el proyecto, además, brindan el impulso y enfoque necesario para su desarrollo exitoso [2].

**Tabla 6** Matriz de Interesados o Stakeholders

MATRIZ GENERAL DE STAKEHOLDERS			
Nombre	Rol	Profesión	Responsabilidad
Ing. Joofre Honores Tapia	Coordinador de carrera / Docente / Gerente de proyecto	Ingeniero de Sistemas	Limitaciones del sistema y revisión de entregables
Erick Fabricio Arévalo Miranda	Desarrollador	Estudiante	Diseño, planificación, implementación y pruebas
Lic. Gerardo Nieto Sojos	Cliente	Licenciado en Cultura Física	Requisitos del sistema, satisfacción de la aplicación

Fuente: Elaboración propia

#### 2.4.1.4. Requerimientos

La recopilación de los requisitos se llevó a cabo mediante una entrevista de tipo verbal directamente con el dueño de la academia, quién manifestó lo que requería la aplicación web. A continuación, se detalla los requerimientos funcionales y no funcionales obtenidos.

##### 2.4.1.4.1. Requerimientos funcionales

Son todas las actividades y comportamientos que el software debe cumplir, además, deben ser capaces de describir la interacción que tiene el sistema con los usuarios finales que accederán a él [2]. En la **tabla 7** se presentan los requisitos funcionales respectivos.

**Tabla 7** Requerimientos Funcionales

REQUERIMIENTOS FUNCIONALES (RF)	
Código	Descripción
RF-01	El sistema permitirá el registro de usuarios
RF-02	El sistema permitirá la autenticación de usuarios
RF-03	Gestión de alumnos
RF-04	Gestión de eventos
RF-05	El sistema permitirá la creación de llaves de competencia de un evento en particular
RF-06	Gestión de Usuarios

Fuente: Elaboración propia

#### 2.4.1.4.2. Requerimientos no funcionales

Son aquellos que se relacionan con las características de la propia aplicación, como el rendimiento, la usabilidad, la velocidad de respuesta y la seguridad [2]. A continuación, se presentan los requerimientos no funcionales.

**Tabla 8** Requerimientos no funcionales

REQUERIMIENTOS NO FUNCIONALES (RNF)	
Código	Descripción
RNF-01	Base de datos estructurada correctamente
RNF-02	La aplicación web debe ser responsiva
RNF-03	La aplicación web debe ser usable, fácil de entender

RNF-04	La aplicación web debe ser rápida al momento de agregar datos
--------	---

Fuente: Elaboración propia

## 2.4.2. Fase de planificación

Es la fase que tiene mayor influencia en el desarrollo del proyecto, en ella se organizan los recursos, tiempo y actividades relevantes y la gestión de los mismos para reducir costes y retrasos en el proceso de desarrollo [2].

### 2.4.2.1. Historias de usuarios

Son descripciones breves y sencillas de la funcionalidad del sistema y la manera en que el usuario interactúa con el mismo, además, se establecen basándose en los requerimientos funcionales [50]; en las siguientes tablas se detalla cada una de las historias de usuarios necesarias para la elaboración de la aplicación.

Tabla 9 Historia de usuario N°1 - Registro al sistema

Historia de Usuario	
Número: 1	Usuarios: Usuario
Nombre de historia: Registro de usuarios	
Prioridad en negocio: Alta	Riesgo de desarrollo: Alta
Requerimiento funcional: RF-01	Iteración asignada: 1
Programador responsable: Erick Fabricio Arévalo Miranda	
Descripción: Los usuarios nuevos deberán cumplir con el registro de su usuario para poder acceder a la aplicación.	
Observaciones: Para poder registrarse deberán hacerlo con un correo electrónico válido.	

Fuente: Elaboración propia

La aplicación web debe permitir la creación de un usuario nuevo para proceder con la autenticación con los datos previamente ingresados.

**Tabla 10** Historia de usuario N°2 - Ingreso al sistema

Historia de Usuario	
Número: 2	Usuarios: Administrador, usuario
Nombre de historia: Ingreso al sistema	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Requerimiento funcional: RF-02	Iteración asignada: 1
Programador responsable: Erick Fabricio Arévalo Miranda	
Descripción: El usuario tanto administrador como normal deben ingresar sus credenciales para acceder a la aplicación web.	
Observaciones: Para el logueo del usuario deben ingresar su correo y contraseña correctos-	

**Fuente:** Elaboración propia

Todos los usuarios creados incluido el administrador deben ingresar correctamente sus credenciales previamente creadas para poder acceder a la aplicación web.

**Tabla 11** Historia N°3 - Gestión de alumnos

Historia de Usuario	
Número: 3	Usuarios: Administrador
Nombre de historia: Gestión de alumnos	
Prioridad en negocio: Alta	Riesgo en desarrollo: Media
Requerimiento funcional: RF-03	Iteración asignada: 1

Programador responsable: Erick Fabricio Arévalo Miranda
Descripción: El usuario administrador podrá crear, editar y eliminar a los alumnos
Observaciones: Los usuarios normales solo podrán visualizar la información de los alumnos.

Fuente: Elaboración propia

El sistema permitirá que solo el usuario administrador pueda crear, editar o eliminar los alumnos que forman parte de la academia; los usuarios registrados serán limitados a visualizar la información de los alumnos sin realizar ninguna operación adicional.

Tabla 12 Historia de usuario N°4 - Gestión de eventos

Historia de Usuario	
Número: 4	Usuarios: Administrador
Nombre de historia: Gestión de eventos	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Requerimiento funcional: RF-04	Iteración asignada: 1
Programador responsable: Erick Fabricio Arévalo Miranda	
Descripción: El administrador podrá crear, editar o eliminar los eventos internos o externos a la academia.	
Observaciones: La creación de eventos no se limita a eventos internos, también se pueden publicar eventos que realicen otras academias o eventos nacionales a manera de información. Solamente el usuario administrador podrá realizar esta operación.	

Fuente: Elaboración propia

El usuario administrador es el único que podrá crear, editar o eliminar eventos, estos eventos pueden ser competencias que va a realizar la academia como campeonatos que vaya a llevar a cabo otra escuela o aquellos que se desarrollen de manera nacional; cabe resaltar que estos últimos serán netamente informativos.

**Tabla 13** Historia de usuario N°5 - Creación de llaves

Historia de Usuario	
Número: 5	Usuarios: Administrador
Nombre de historia: Creación de llaves de competencia	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Requerimiento funcional: RF-05	Iteración asignada: 1
Programador responsable: Erick Fabricio Arévalo Miranda	
Descripción: La aplicación permitirá crear las llaves de competencia de un evento que realice la academia.	
Observaciones: La creación de las llaves de competencia se realizará únicamente en los eventos de competencia que vaya a realizar la academia.	

**Fuente:** Elaboración propia

Las llaves de competencia se realizarán únicamente cuando sea un evento de competencia que realice la academia, el procedimiento es el siguiente: el administrador inscribirá a los deportistas que formarán parte de este evento y automáticamente se crearán los pools de pelea para el desarrollo del mismo.

**Tabla 14** Historia de usuario N°6 - Gestión de usuarios

Historia de Usuario	
Número: 6	Usuarios: Usuario
Nombre de historia: Gestión de usuarios	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Requerimiento funcional: RF-06	Iteración asignada: 1
Programador responsable: Erick Fabricio Arévalo Miranda	
Descripción: Los usuarios no admin podrán realizar operaciones como registrarse y autenticarse, visualizar información sobre los alumnos pertenecientes a la academia y tener acceso a los eventos que se estén realizando.	
Observaciones: El usuario no admin no podrá realizar ninguna operación de creación, edición o eliminación que tenga que ver con la gestión de alumnos y eventos.	

**Fuente:** Elaboración propia

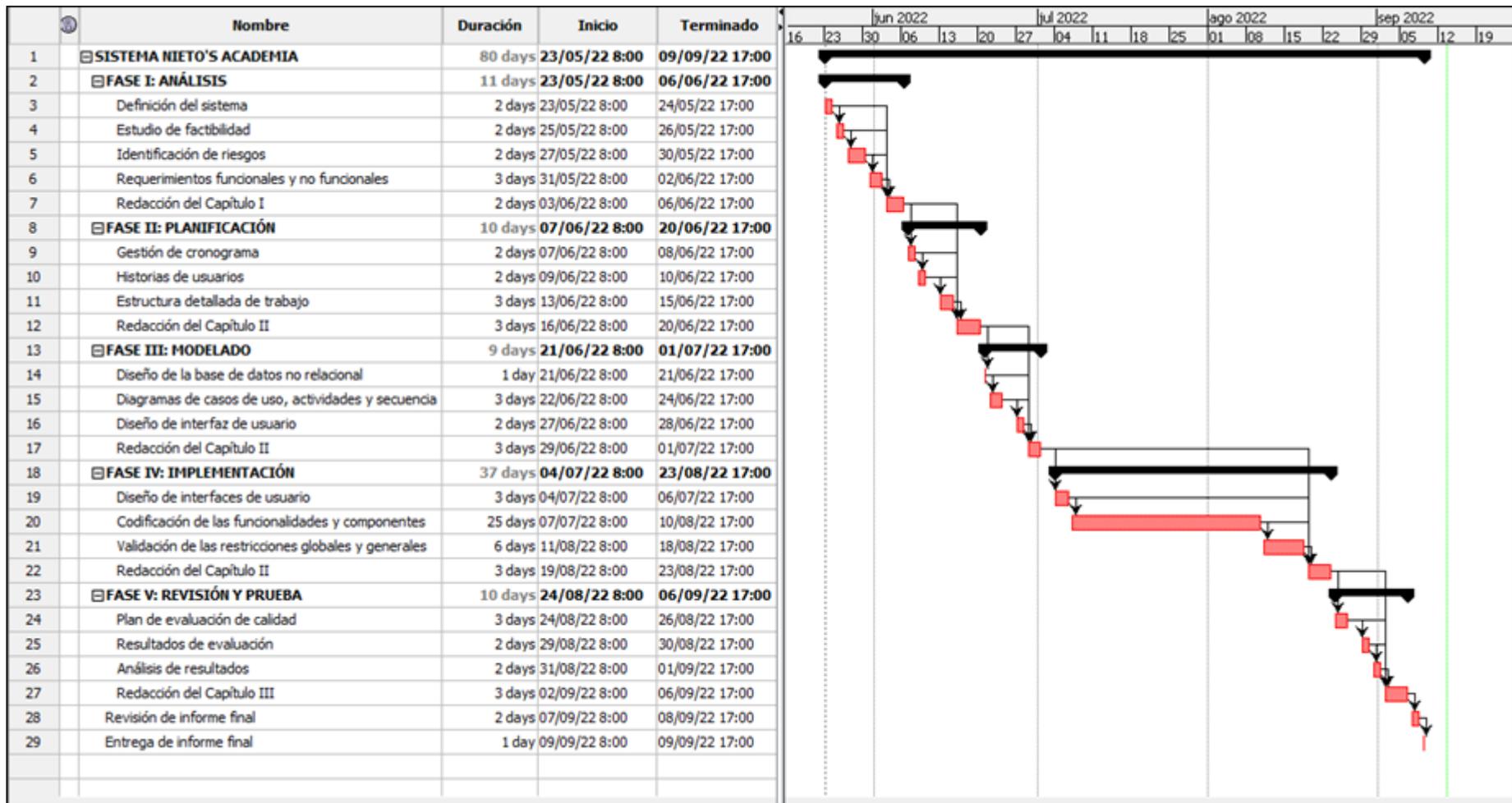
El usuario no admin tiene únicamente acceso de forma informativa a las tareas que realiza la aplicación, mientras que el usuario administrador es el único que puede manipular en su totalidad el sistema.

#### **2.4.2.2. Gestión del cronograma**

El cronograma de actividades detalla el proceso de desarrollo de la aplicación web, desde la fase de análisis con la toma de requisitos hasta la implementación, revisión y pruebas del mismo.

A continuación, se presenta el cronograma de actividades junto a las fechas establecidas y su respectivo diagrama de Gantt.

Ilustración 5 Cronograma de actividades



Fuente: Elaboración propia

En la ilustración 5, se puede observar el cronograma de actividades con su respectivo diagrama de Gantt basado en las etapas de la metodología “SWIRL” con un tiempo estimado de 80 días para la finalización de la aplicación web.

### **2.4.2.3. Gestión de riesgos**

El propósito de la Gestión de Riesgos es identificar, controlar y eliminar las fuentes de riesgo antes de que comiencen a afectar el logro de los objetivos del proyecto; de esta manera, se puede evaluar y estimar el impacto de los posibles riesgos para luego desarrollar planes de contingencia y actuar proactivamente para mitigarlos en caso de ser necesario [51]; por lo tanto, se debe tomar en cuenta los siguientes aspectos para realizar el respectivo análisis de riesgos:

- Comenzar antes del trabajo técnico.
- Clasificar los riesgos según el impacto.
- Identificar los riesgos potenciales y valorar su probabilidad de impacto en el proyecto.

En la presente propuesta tecnológica la gestión de riesgos tiene el objetivo de minimizar los acontecimientos negativos que se puedan suscitar a lo largo del desarrollo del proyecto; el análisis de los riesgos se los encuentra detallados en el ANEXO A: Análisis de riesgos.

### **2.4.3. Fase de modelado**

La fase de modelado describe los diversos modelos de base de datos que la aplicación web necesita integrar para su funcionamiento y la estructura lógica del sistema, en esta aplicación se utilizará la base de datos no estructurada o no relacional.

#### **2.4.3.1. Diseño de base de datos no relacional**

Para el manejo de los datos que utilizará la aplicación web se empleará una distribución de base de datos no relacional por su escalabilidad, flexibilidad y alta disponibilidad; debido a que este sistema manejará un alto volumen de información por los procesos que realiza la academia se decidió aplicar este tipo de estructura.

A continuación, se muestra el esquema de la base de datos que en lugar de utilizar tablas como las estructura SQL se usan colecciones en formato parecido a un JSON.



**Ilustración 8** Modelo de base de datos NoSQL Módulo Eventos

```
"eventos": [  
  {  
    "title": "Open Machala 2022",  
    "lugar": "Coliseo 3000",  
    "fecha": "2022-09-10T00:00:00.000Z",  
    "ciudad": "Machala",  
    "user": {  
      "_id": "63065f761dd82d9717efcf0c",  
      "name": "Administrador"  
    },  
    "id": "630c664f421f1b272db7b0cb"  
  },  
],
```

**Fuente:** Elaboración propia

En la **ilustración 8**, se puede observar el módulo de eventos de la base de datos del sistema, el mismo que ayudará a manipular la información de todos los eventos realizados por la academia.

### 2.4.3.2. Diseño de interfaces

En esta sección se utilizó la herramienta de bosquejo de interfaces Balsamiq Mockups, la cual permite tener una idea real del aspecto final que tendrá la aplicación web a desarrollar. A continuación, se presenta cada una de las interfaces con las que contará el sistema.

## Login – Registro

**Ilustración 9** Diseño de Login-Registro

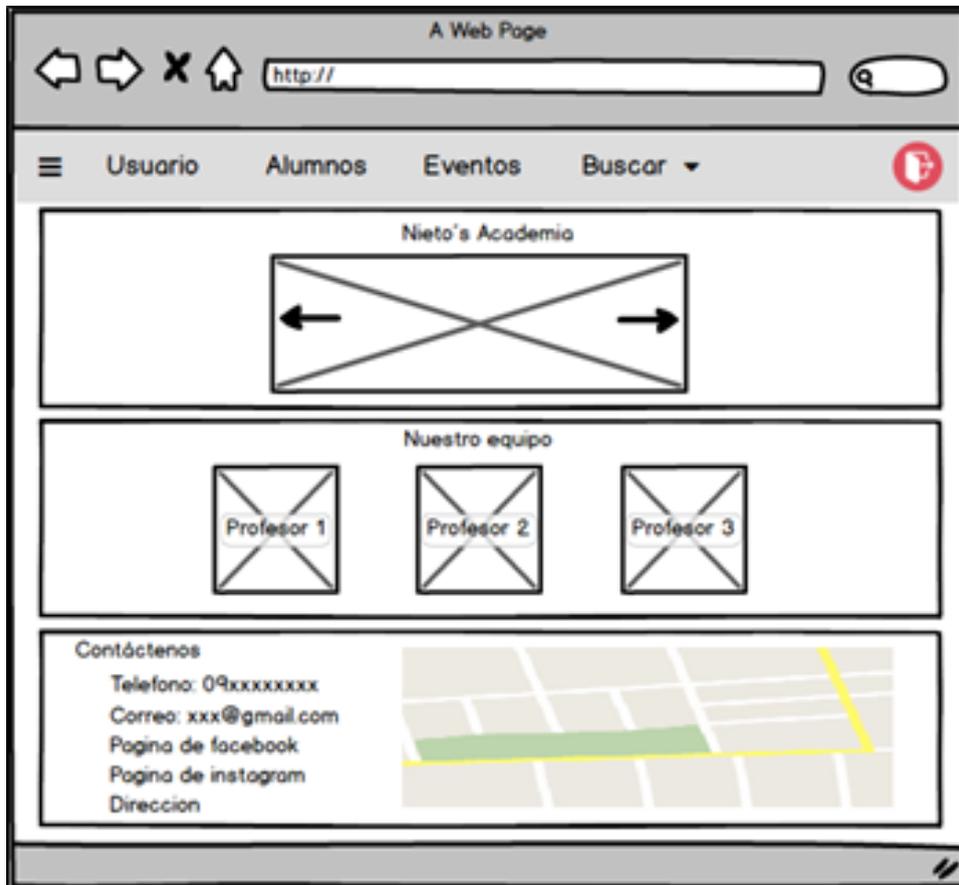


**Fuente:** Elaboración propia

En la ilustración 9, se puede observar la página principal y logeo del sistema web, el usuario podrá ingresar a la aplicación con su correo y contraseña si es que ha cumplido con el registro previo en la plataforma.

## Página Inicio

Ilustración 10 Diseño de la Página de Inicio

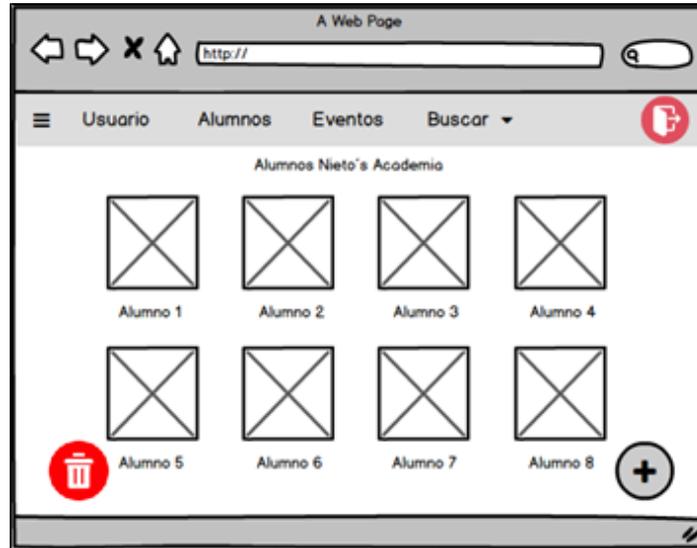


Fuente: Elaboración propia

En la **ilustración 10**, se observa la página de inicio de la aplicación donde se encuentra el logotipo de la academia, un pequeño carrusel con imágenes de las clases y alumnos; el equipo de trabajo; información y la ubicación de la escuela.

## Alumnos

Ilustración 11 Diseño de la Página de Alumnos

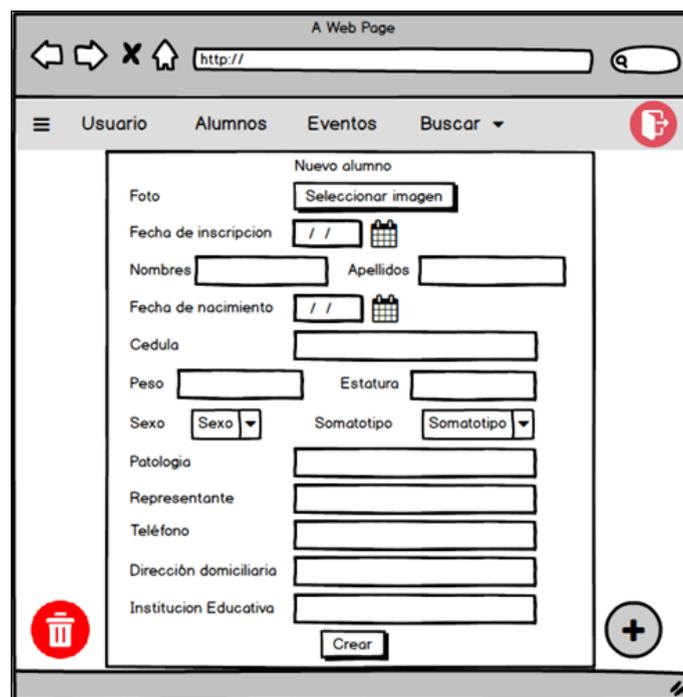


Fuente: Elaboración propia

Se puede observar en la **ilustración 11**, la gestión de alumnos de la academia donde se encontrarán todos los estudiantes y la información de cada uno de ellos, de igual manera la aplicación contará con las operaciones de creación, edición y eliminación de los registros de los mismos.

## Creación de alumnos

Ilustración 12 Diseño de la creación de un alumno

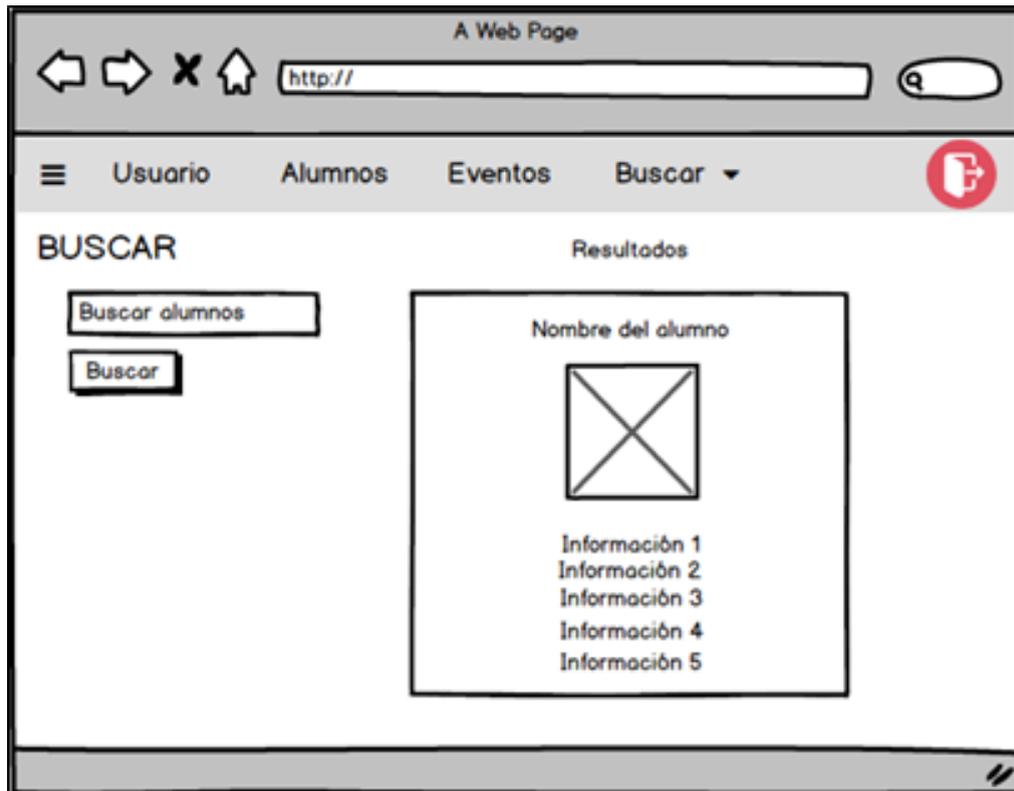


Fuente: Elaboración propia

En la **ilustración 12**, se puede observar el formulario de creación de alumnos, el mismo que servirá para editar la información de los estudiantes, el añadir una imagen es opcional, de no hacerlo, se le asignará una por defecto; una vez registrado el alumno su información será añadida a la página principal mencionada anteriormente en la **ilustración 11**.

## Buscador

Ilustración 13 Diseño del buscador de alumnos

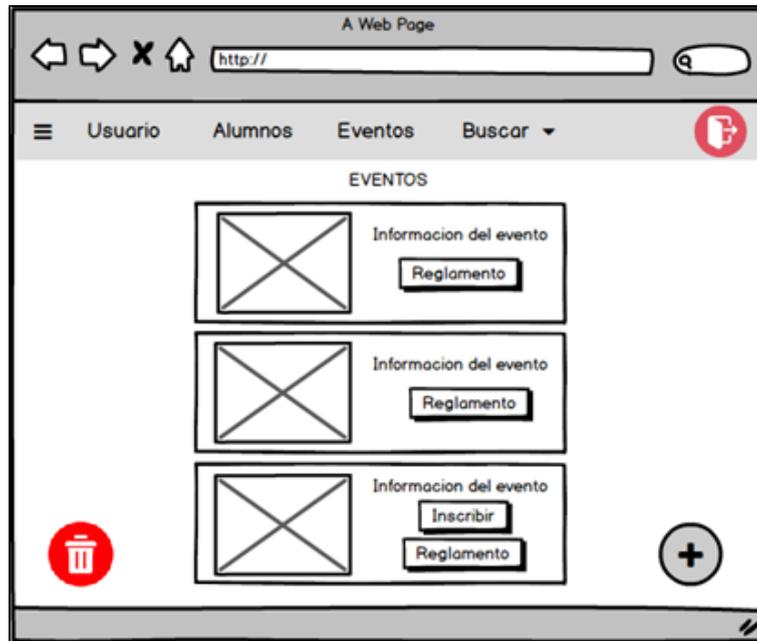


Fuente: Elaboración propia

En la ilustración 13, se puede observar el módulo de búsqueda, el cual ha sido añadido con la finalidad de facilitar al administrador de la aplicación encontrar rápidamente la información de un alumno.

## Eventos

Ilustración 14 Diseño de la Página de Eventos

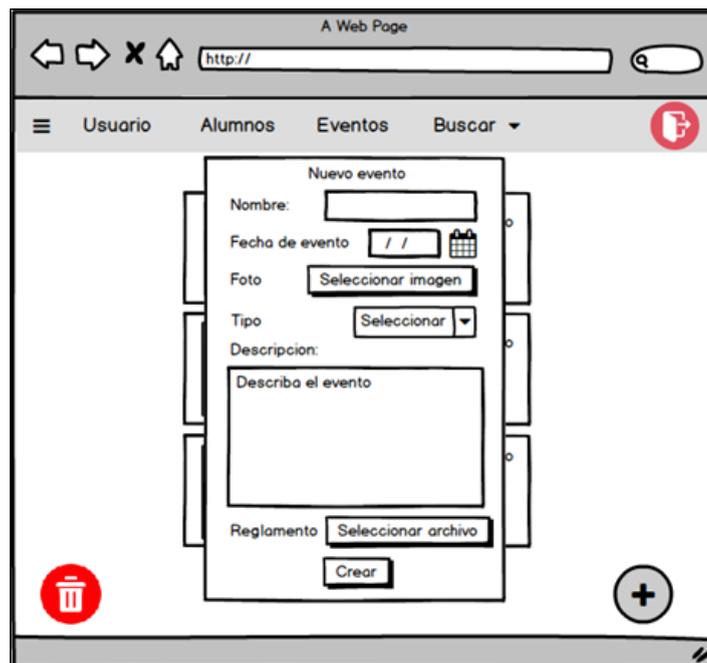


Fuente: Elaboración propia

Se puede observar en la ilustración 14, la gestión de los diferentes eventos ya sea los propios de la academia o aquellos que realicen otras escuelas o los que son de carácter nacional, los cuales serán netamente informativos.

## Creación de eventos

Ilustración 15 Diseño de la creación de un evento

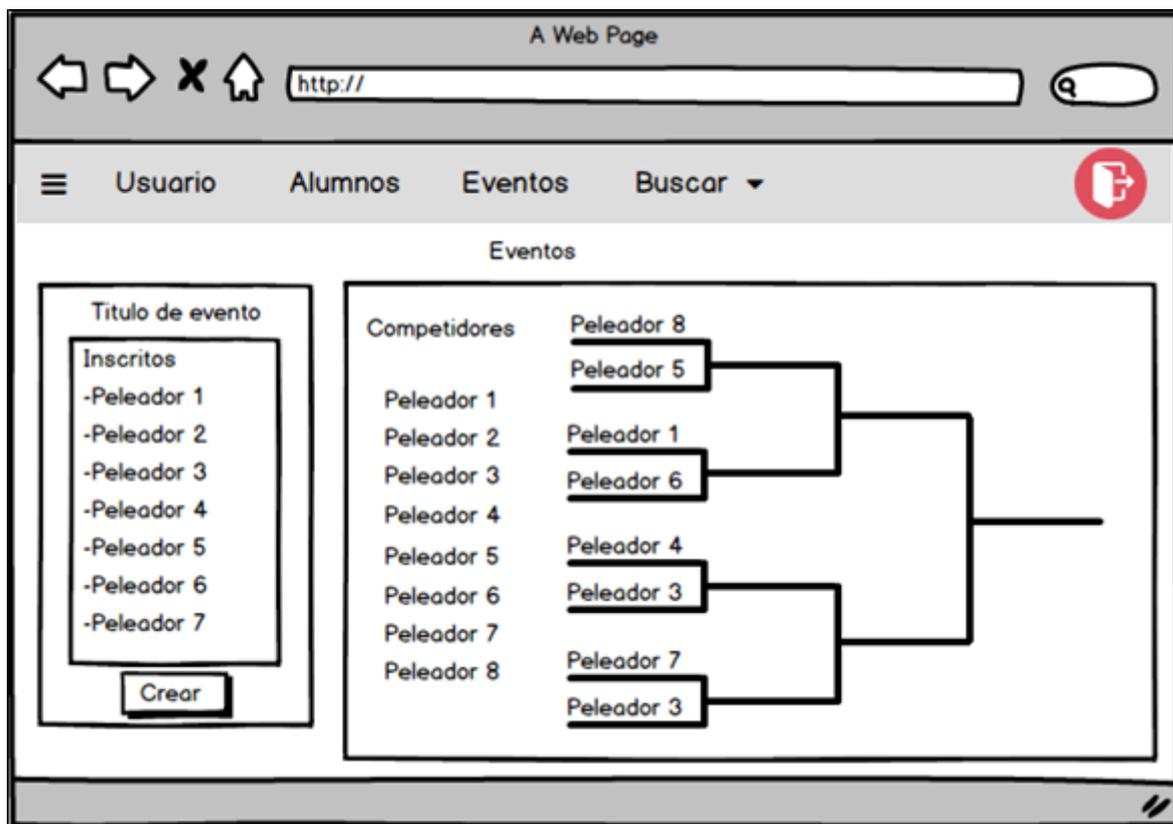


Fuente: Elaboración propia

En la **ilustración 15**, se puede observar el formulario para la creación de un evento, el mismo que trae apartados como el nombre, la fecha que se realizará el campeonato, la imagen, descripción y el reglamento; un apartado a destacar es el tipo de evento “Propio o informativo” debido a que ayudará en la creación de las llaves de competencia, cuando se haya creado correctamente el evento se añadirá a la página principal mencionada en la **ilustración 14**.

## Llaves de Campeonato

**Ilustración 16** Diseño de la creación de pools de pelea



**Fuente:** Elaboración propia

En la **ilustración 16**, se puede observar la creación de los pools o llaves de competencia, cabe mencionar que este proceso solo se podrá realizar en eventos que sean propios de la academia; en la imagen se puede distinguir dos áreas, en el lado izquierdo se encuentran los competidores que formarán parte de dicho campeonato y en el lado derecho se visualiza las llaves de peleas creadas de forma aleatoria y automática conformadas por los participantes previamente inscritos.

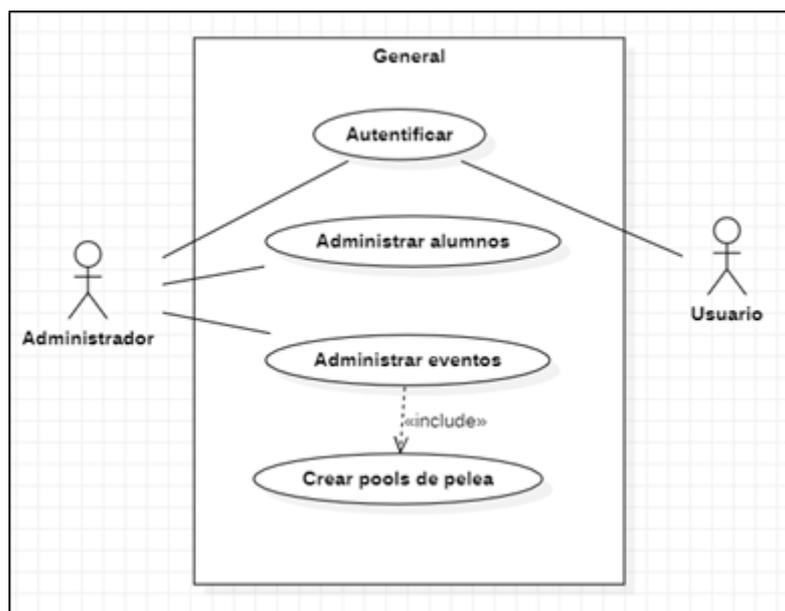
### 2.4.3.3. Diseño de diagramas UML

Los diagramas UML (Lenguaje Modelado Unificado) facilitan la visualización de procesos, clases, propiedades y objetos que están presentes en un proyecto mediante el uso de diferentes gráficos que permiten comprender cómo interactúa el usuario con el sistema [52]. En la siguiente sección se presenta una serie de diagramas que ayudan a tener una mejor interacción con la aplicación a desarrollar.

#### 2.4.3.3.1. Diagramas de caso de uso

Los diagramas de casos de uso nos permiten comprobar que se han cumplido satisfactoriamente y con calidad todos los requisitos que debe cumplir la aplicación resultante [53]. A continuación, se muestran los diagramas de caso de uso del sistema.

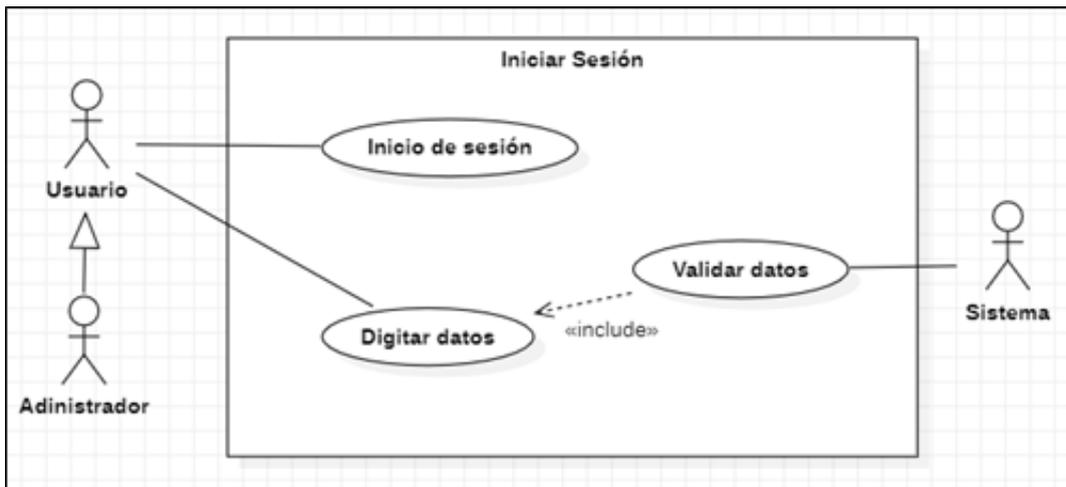
Ilustración 17 Diagrama de caso de uso - General



Fuente: Elaboración propia

En la **ilustración 17**, se observa que el administrador es el único que tiene control total sobre los procesos que realiza la aplicación web como la administración de alumnos, de eventos y la creación de llaves de competencia mientras que el usuario registrado sólo tendrá acceso visual dentro del sistema.

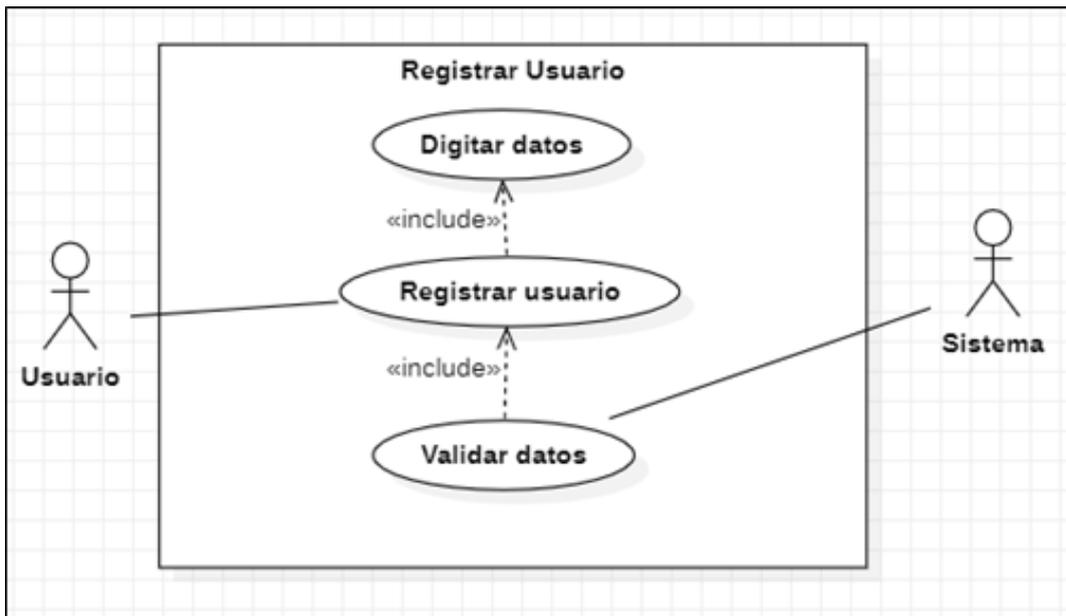
**Ilustración 18** Diagrama de caso de uso - Iniciar Sesión



Fuente: Elaboración propia

En la **ilustración 18**, se observa que todos los usuarios pueden ingresar a la aplicación web mediante credenciales que deben ser validadas por el sistema.

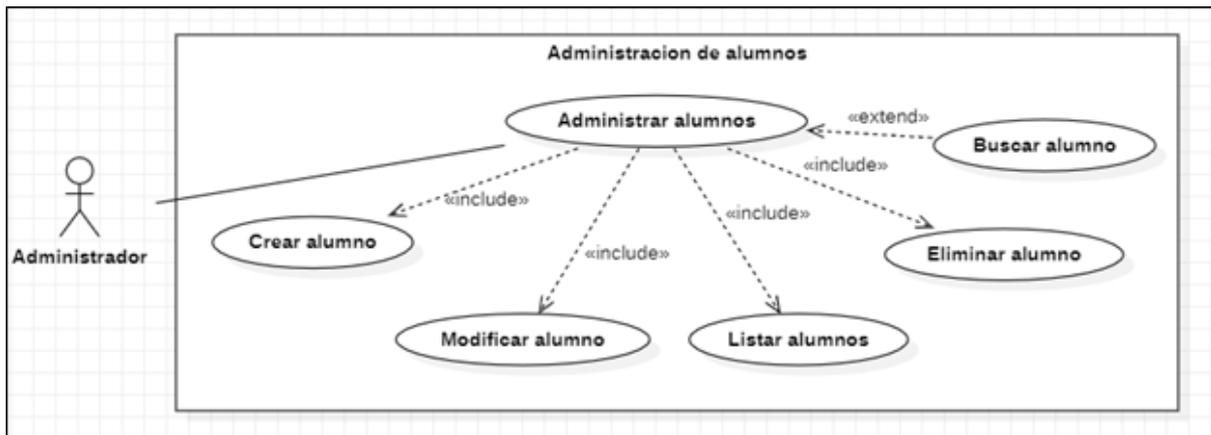
**Ilustración 19** Diagrama de caso de uso - Registrar Usuario



Fuente: Elaboración propia

En la **ilustración 19**, se observa como un usuario nuevo debe cumplir con el apartado de registro para poder ingresar al sistema completando todos los campos que solicite la aplicación.

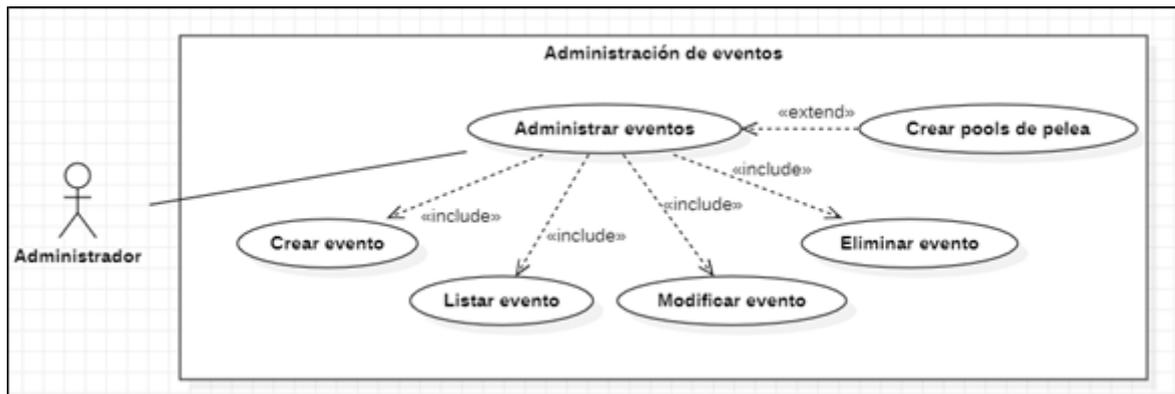
**Ilustración 20** Diagrama de caso de uso - Administración de Alumnos



Fuente: Elaboración propia

En la **ilustración 20**, se puede observar cómo el usuario administrador puede realizar las tareas de gestión de alumnos como lo son crear, modificar, listar y eliminar estudiantes.

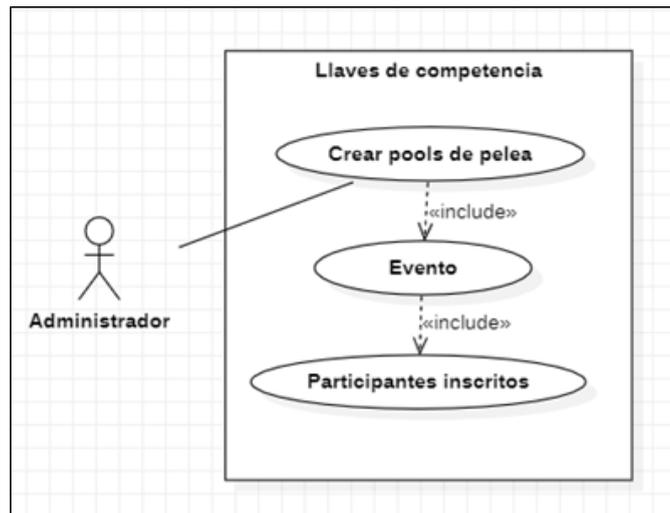
**Ilustración 21** Diagrama de caso de uso - Administración de Eventos



Fuente: Elaboración propia

En la **ilustración 21**, se observa que el administrador tiene los privilegios para gestionar eventos realizando tareas CRUD como crear, listar, modificar y eliminar eventos; adicionalmente existe la tarea denominada crear pools de pelea, la cual puede ser accedida solo por eventos realizados propiamente por la academia.

**Ilustración 22** Diagrama de caso de uso - Creación de pools de pelea



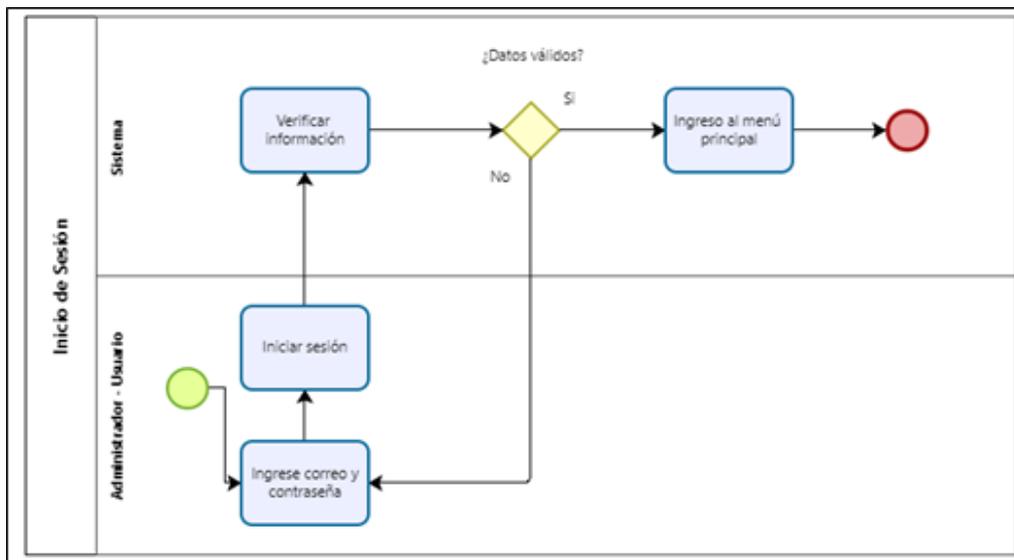
Fuente: Elaboración propia

En la **ilustración 22**, se observa que únicamente el administrador podrá crear automáticamente los pools o llaves de competencia de un campeonato, para realizar este proceso debe de existir un evento que sea propio de la academia y a su vez tienen que estar inscritos competidores.

#### 2.4.3.3.2. Diagramas de actividades

Son aquellos que detallan de principio a fin el intercambio de procesos o actividades en cada módulo de un sistema, son muy similares a los diagramas de flujo con la diferencia de que interactúan varios actores. A continuación, se presentan los diagramas con las actividades que realiza la aplicación.

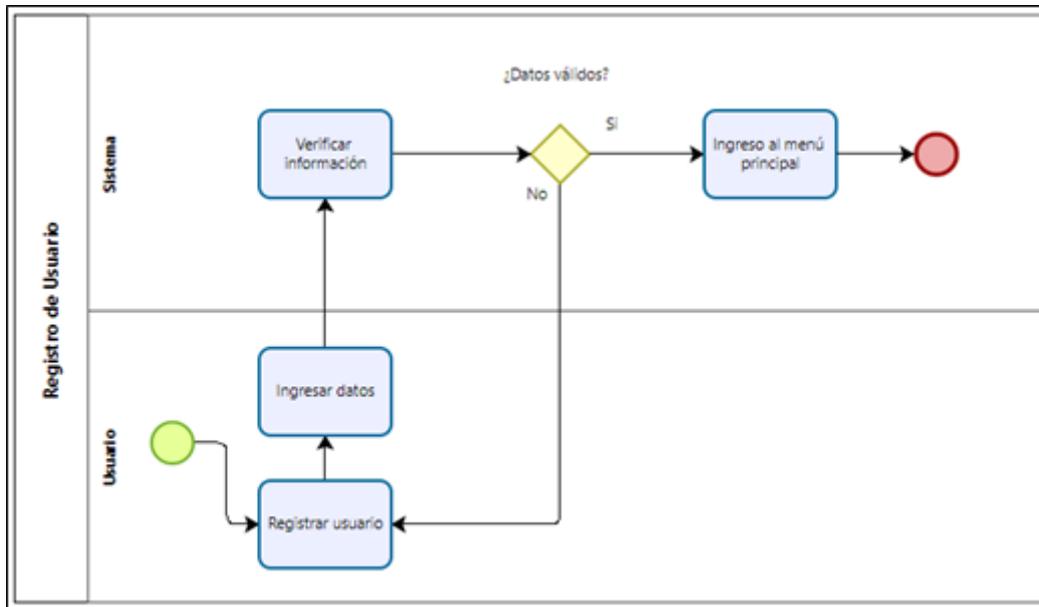
**Ilustración 23** Diagrama de actividades - Inicio de Sesión



Fuente: Elaboración propia

En la **ilustración 23**, se puede observar las validaciones que realiza la aplicación para que usuario o el administrador puedan acceder al sistema.

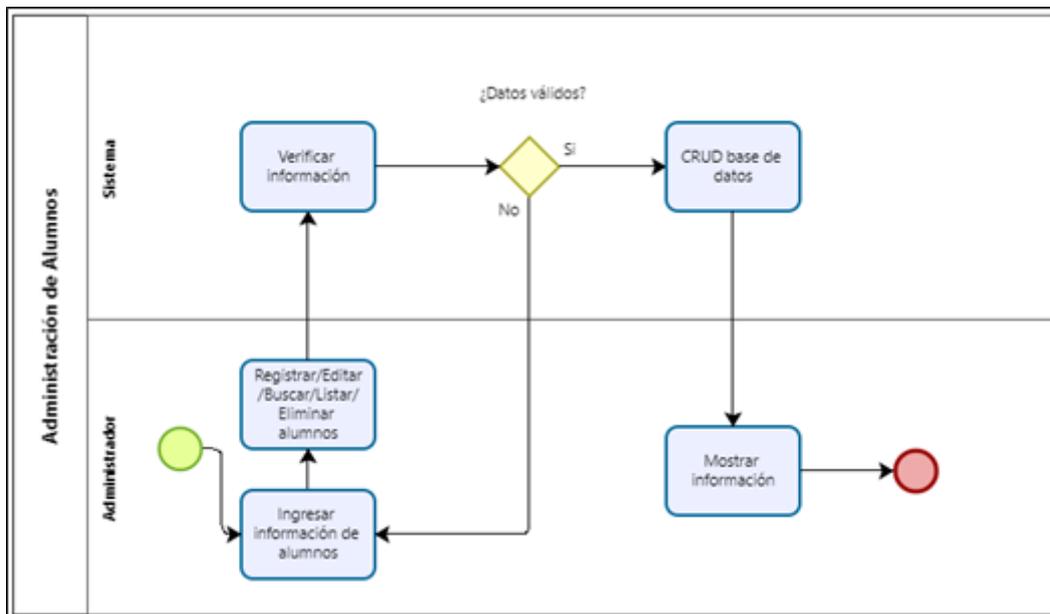
**Ilustración 24** Diagrama de actividades - Registro de Usuario



Fuente: Elaboración propia

En la **ilustración 24**, se puede observar las actividades que se deben realizar para que un usuario pueda ser registrado en el sistema.

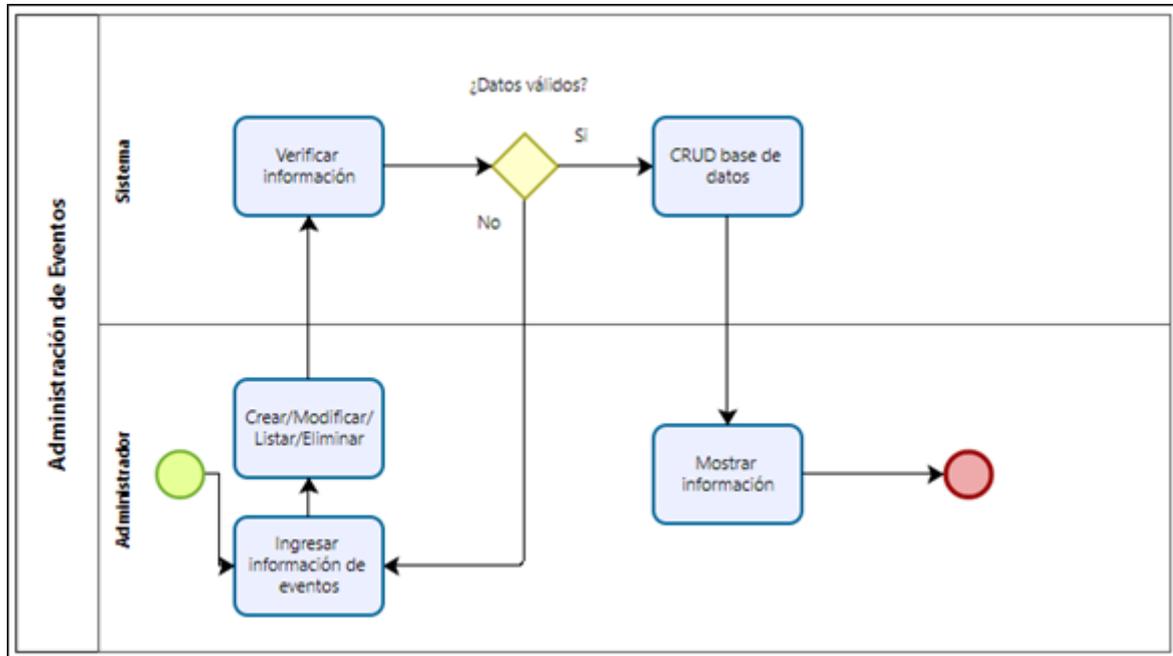
**Ilustración 25** Diagrama de actividades - Administración de Alumnos



Fuente: Elaboración propia

La **ilustración 25**, detalla el proceso que lleva a cabo el administrador para realizar la gestión de alumnos, ya sea que se ingrese, edite, liste, busque o elimine un estudiante.

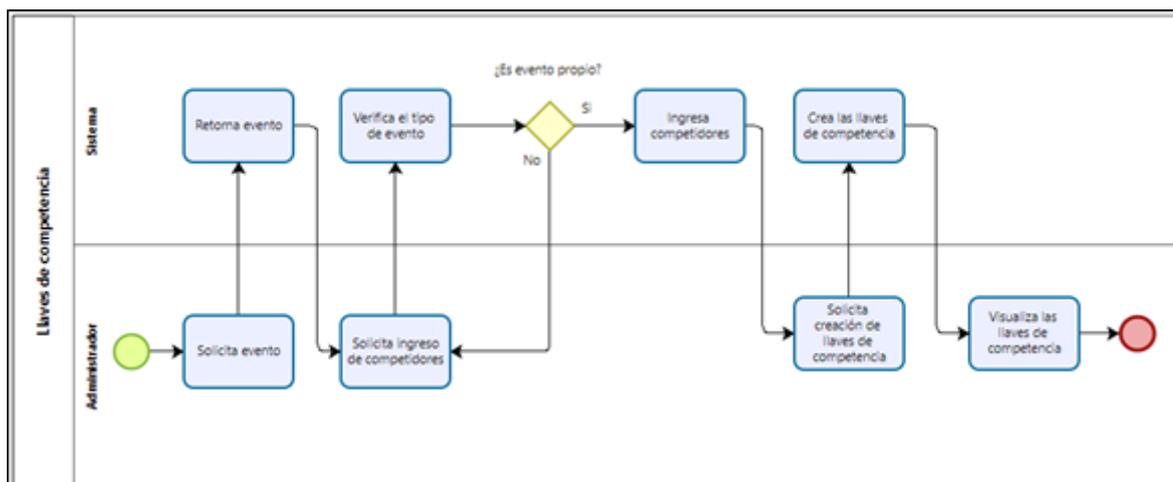
**Ilustración 26** Diagrama de actividades - Administración de Eventos



Fuente: Elaboración propia

La **ilustración 26**, hace referencia a las operaciones CRUD de los eventos que maneja la academia, es decir la administración de los mismos.

**Ilustración 27** Diagrama de Actividades - Creación de llaves de competencia



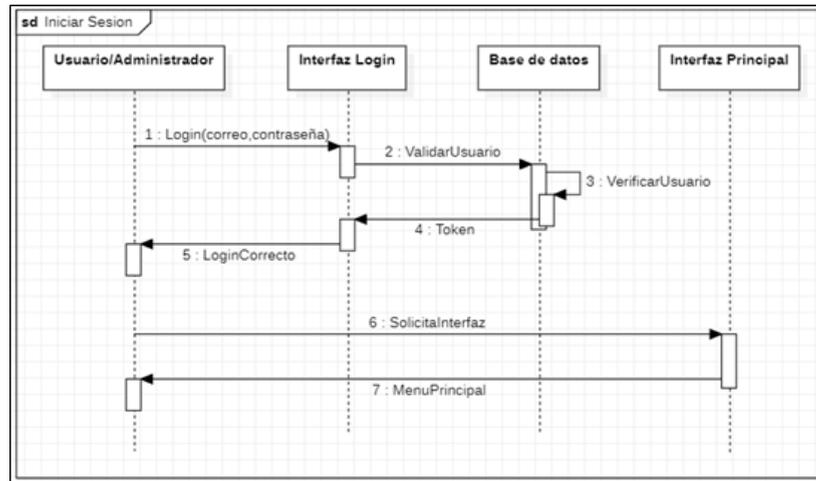
Fuente: Elaboración propia

Como se puede observar en la **ilustración 27**, el diagrama de actividades indica las acciones necesarias para la creación de los pools de pelea.

### 2.4.3.3.3. Diagramas de secuencias

Los diagramas de secuencias son usados para modelar la interacción entre objetos en un sistema, definiendo el tiempo de ejecución de los procesos que intervienen en cada actividad del mismo.

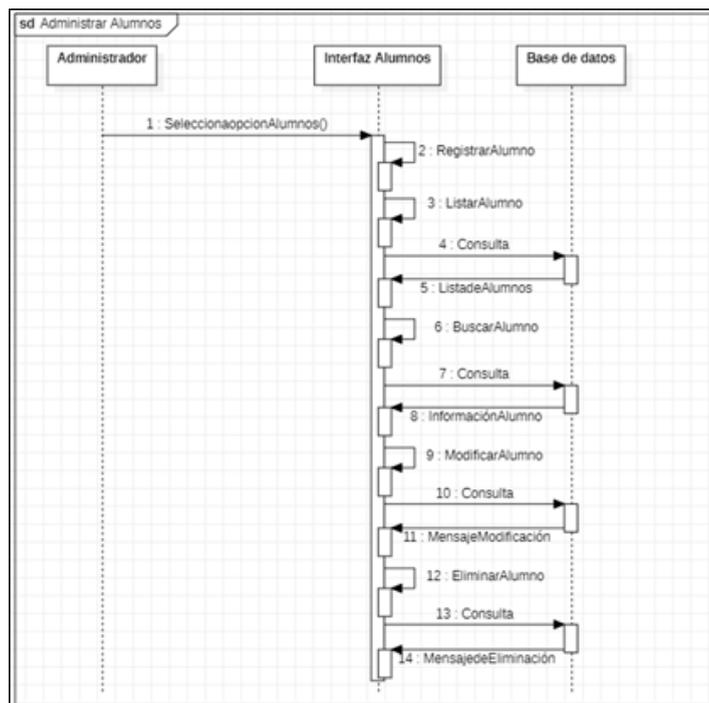
Ilustración 28 Diagrama de secuencia - Inicio de Sesión



Fuente: Elaboración propia

La **ilustración 28**, especifica a detalle el proceso que hace el administrador con la aplicación y el desarrollo interno del sistema para mostrarle el contenido al administrador.

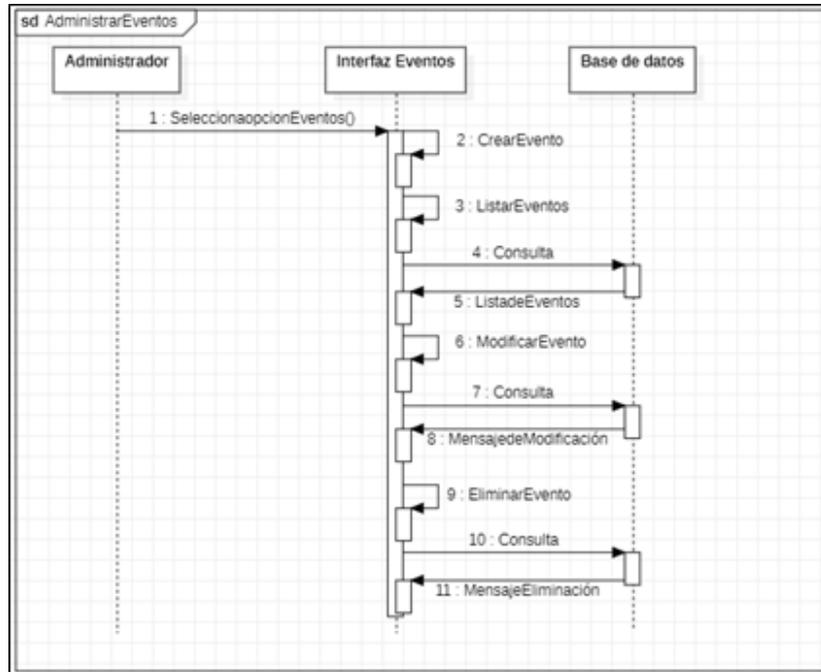
Ilustración 29 Diagrama de secuencia - Administrar Alumnos



Fuente: Elaboración propia

En la **ilustración 29**, se puede observar el proceso que realiza el sistema para manejar la gestión de alumnos permitiendo crear, editar, listar, buscar y eliminar la información de los mismos mediante peticiones a la base de datos.

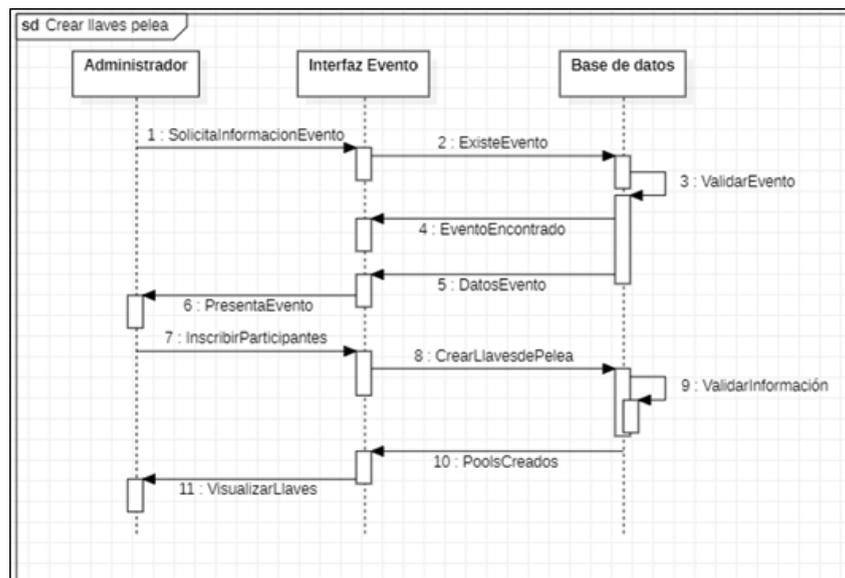
**Ilustración 30** Diagrama de secuencia - Administrar Eventos



Fuente: Elaboración propia

Al igual que el proceso anterior, la **ilustración 30**, nos indica la secuencia realizada por el sistema para administrar los eventos realizados en la academia o fuera de ella a través de peticiones a la base de datos.

**Ilustración 31** Diagrama de secuencia - Crear llaves de pelea



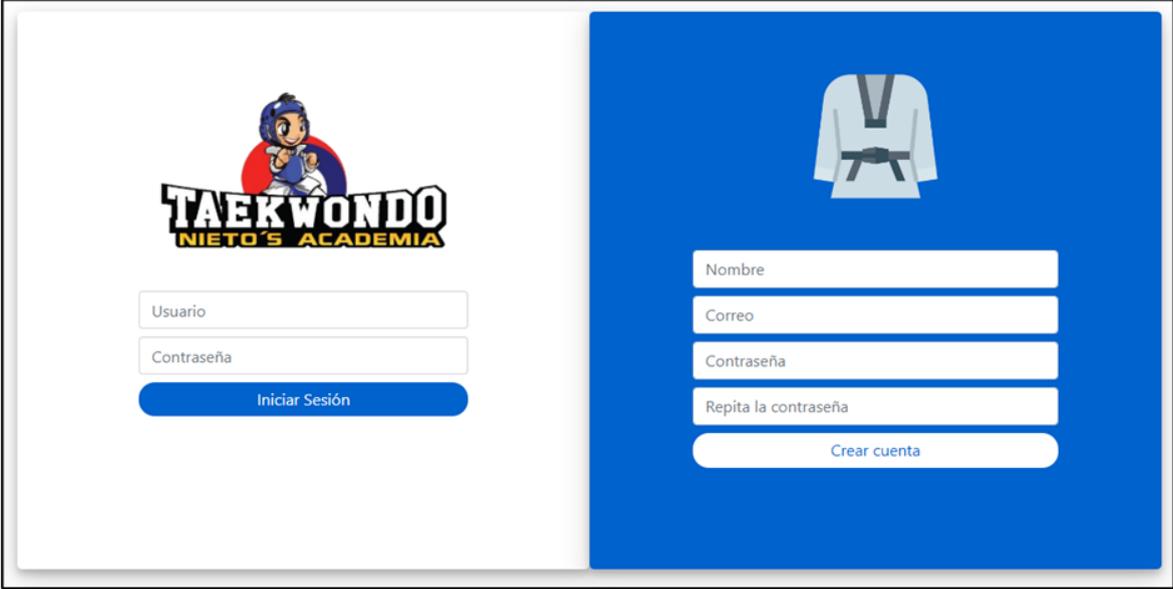
Fuente: Elaboración propia

En la **ilustración 31**, se indica el proceso realizado por el sistema para la creación de los pools o llaves de competencia de los eventos realizados por la academia, a través de peticiones y validaciones a la base de datos.

## 2.5. Ejecución y/o ensamble del prototipo

En el siguiente apartado se presenta la aplicación web ya terminada con sus respectivos módulos de alumnos, eventos y creación de llaves de competencia, utilizando las herramientas y tecnologías mencionadas en los puntos anteriores.

**Ilustración 32** Implementación - Página Login-Registro

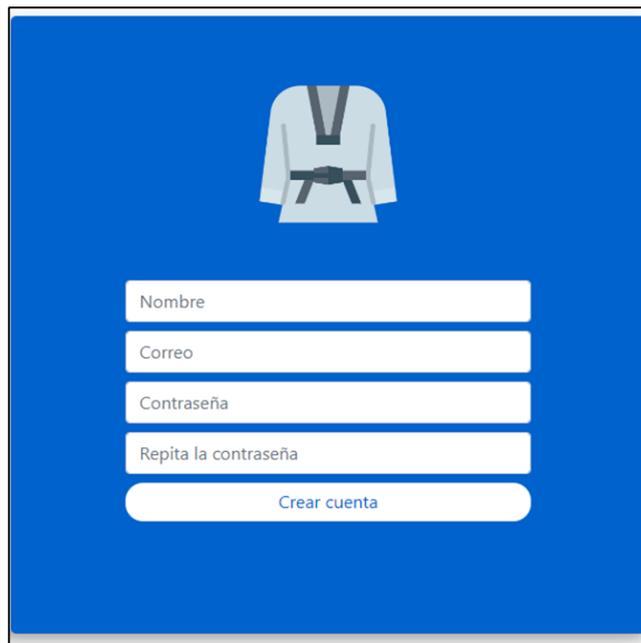


The image shows a web application interface for 'TAEKWONDO NIETO'S ACADEMIA'. The page is split into two main sections. The left section has a white background and features the academy's logo at the top, which includes a cartoon character in a blue and red uniform. Below the logo are two input fields labeled 'Usuario' and 'Contraseña', followed by a blue button labeled 'Iniciar Sesión'. The right section has a blue background and features an illustration of a white Taekwondo gi with a black belt. Below the illustration are four input fields: 'Nombre', 'Correo', 'Contraseña', and 'Repita la contraseña', followed by a white button labeled 'Crear cuenta'.

**Fuente:** Elaboración propia

Como se puede observar en la **ilustración 32**, tenemos la primera vista de la aplicación, donde el usuario tendrá que realizar el registro o login.

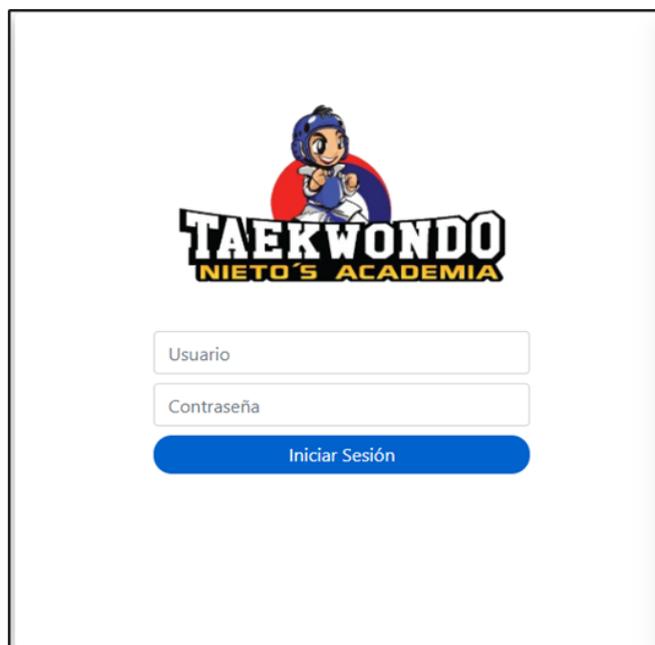
**Ilustración 33** Implementación - Registro



**Fuente:** Elaboración propia

En la **ilustración 33**, observamos el apartado de registro donde un usuario nuevo debe ingresar sus respectivos datos válidos para luego realizar la autenticación.

**Ilustración 34** Implementación - Iniciar Sesión



**Fuente:** Elaboración propia

Se puede observar en la **ilustración 34** el Login de la aplicación web, donde un usuario que previamente se haya registrado podrá proceder a autenticarse para ingresar al sistema.

Ilustración 35 Página Principal

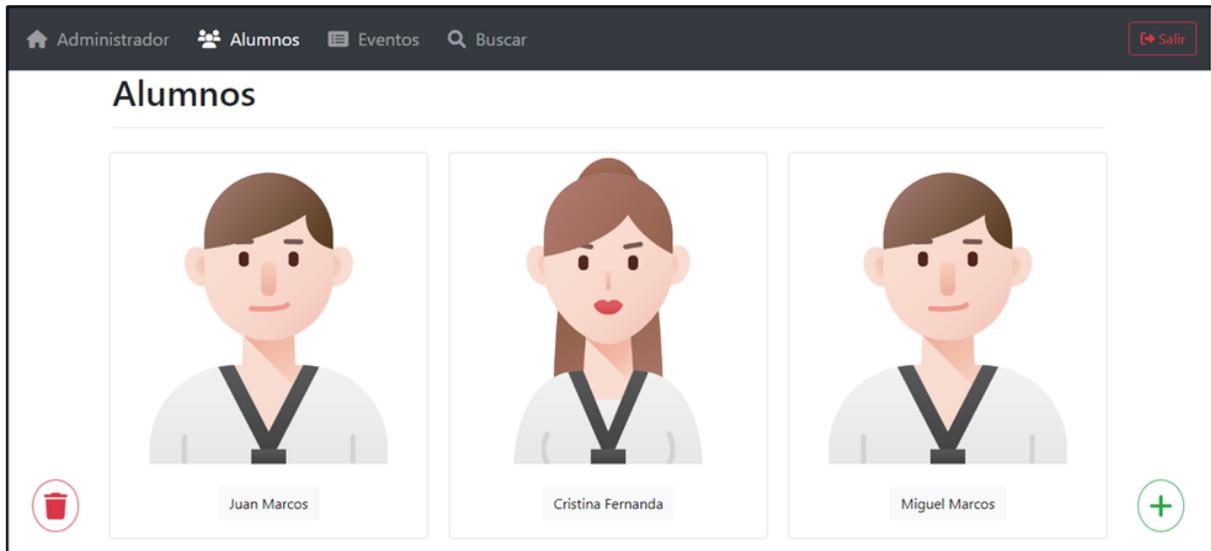


Fuente: Elaboración propia

En la **ilustración 35**, se puede observar la página principal de la aplicación web con información importante respecto a la academia como imágenes de los entrenamientos y actividades que se realizan, información de los maestros, sistema

educativo y un apartado interesante donde se encuentra la ubicación y la forma de contactar con la escuela.

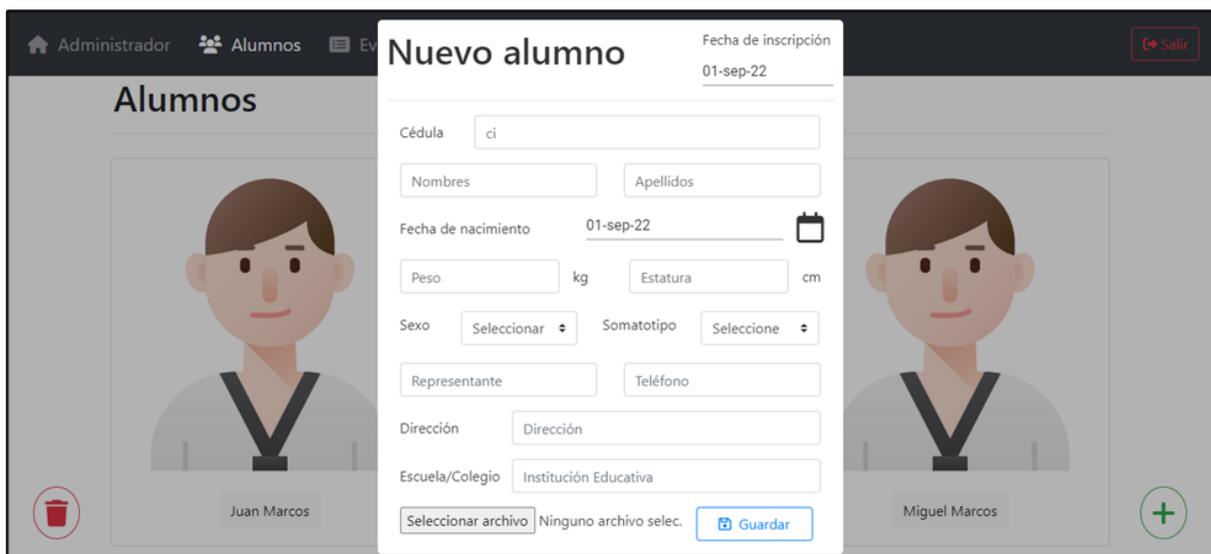
**Ilustración 36** Página Alumnos



**Fuente:** Elaboración propia

La **ilustración 36**, presenta la página de los alumnos, donde se encuentran todos los estudiantes que forman parte de la academia con su imagen y su respectivo nombre, de tal manera que sea fácil para el usuario encontrar a uno en particular; para acceder a la información completa del estudiante se debe hacer un click en el nombre del mismo.

**Ilustración 37** Formulario para crear un alumno



**Fuente:** Elaboración propia

En la **ilustración 37**, se observa el formulario para la creación de un nuevo alumno, en él se puede observar toda la información necesaria para agregarlo; es importante destacar que su diseño se realizó en base a la ficha de inscripción propiamente de la escuela.

**Ilustración 38** Información del Alumno



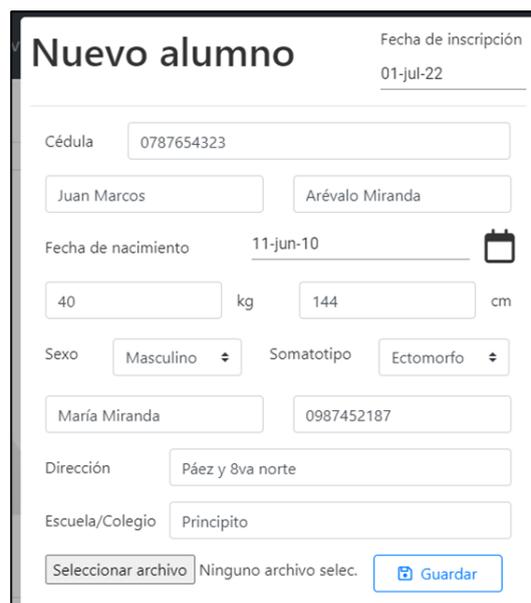
The screenshot shows a web interface with a navigation bar at the top containing 'Administrador', 'Alumnos', 'Eventos', and 'Buscar'. A 'Salir' button is in the top right. The main content area features a student profile for 'Juan Marcos Arévalo Miranda'. On the left is a placeholder for a student photo. To the right, the following information is displayed in a list format:

- Fecha de inscripción: 2022-07-01
- Cédula: 0787654323
- Fecha de nacimiento: 2010-06-11
- Peso: 40 kg
- Estatura: 144 cm
- Somatotipo: Ectomorfo
- Representante: María Miranda
- Teléfono: 0987452187
- Dirección: Paéz y 8va norte

**Fuente:** Elaboración propia

Se puede observar en la **ilustración 38** la página que muestra la información sobre un alumno determinado, para acceder a ella se debe dar click en el nombre del estudiante tal y como se mencionó en la **ilustración 36**; es imprescindible recordar que únicamente el usuario administrador tendrá acceso a dichas características.

**Ilustración 39** Modificar Alumno



The screenshot shows a form titled 'Nuevo alumno' with a 'Fecha de inscripción' field set to '01-jul-22'. The form contains the following fields and values:

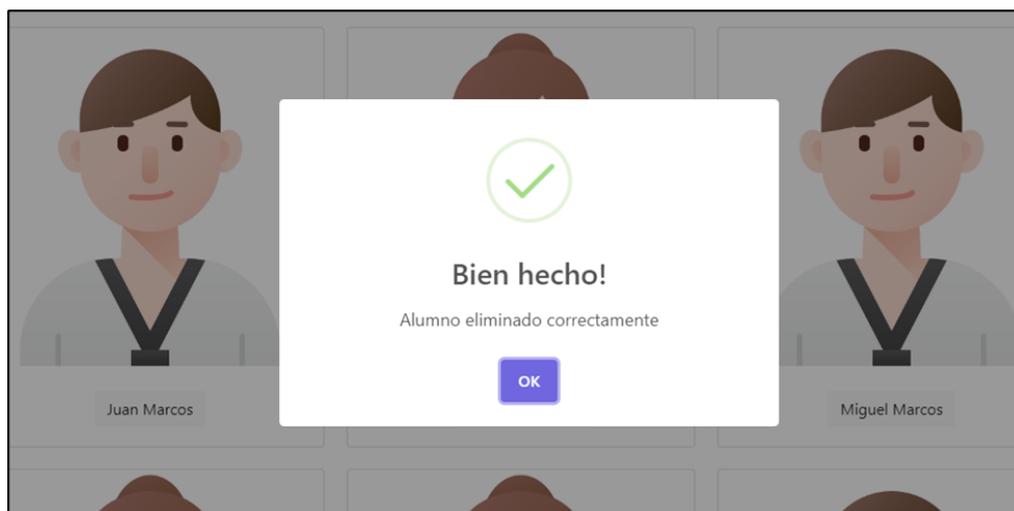
- Cédula: 0787654323
- Nombre: Juan Marcos
- Apellido: Arévalo Miranda
- Fecha de nacimiento: 11-jun-10
- Peso: 40 kg
- Estatura: 144 cm
- Sexo: Masculino
- Somatotipo: Ectomorfo
- Representante: María Miranda
- Teléfono: 0987452187
- Dirección: Paéz y 8va norte
- Escuela/Colegio: Principito

At the bottom, there is a file upload section with 'Seleccionar archivo', 'Ninguno archivo selec.', and a 'Guardar' button.

**Fuente:** Elaboración propia

Como se puede observar en la **ilustración 39**, se muestra el formulario para realizar la respectiva actualización de un alumno en particular, para poder acceder a esta función se debe dirigir a la lista de todos los alumnos mencionada en la **ilustración 36** para luego dar doble click en la imagen del deportista solicitado de tal manera que se despliegue el formulario con la información del mismo, de esa forma se puede modificar el apartado que se necesite hacerlo.

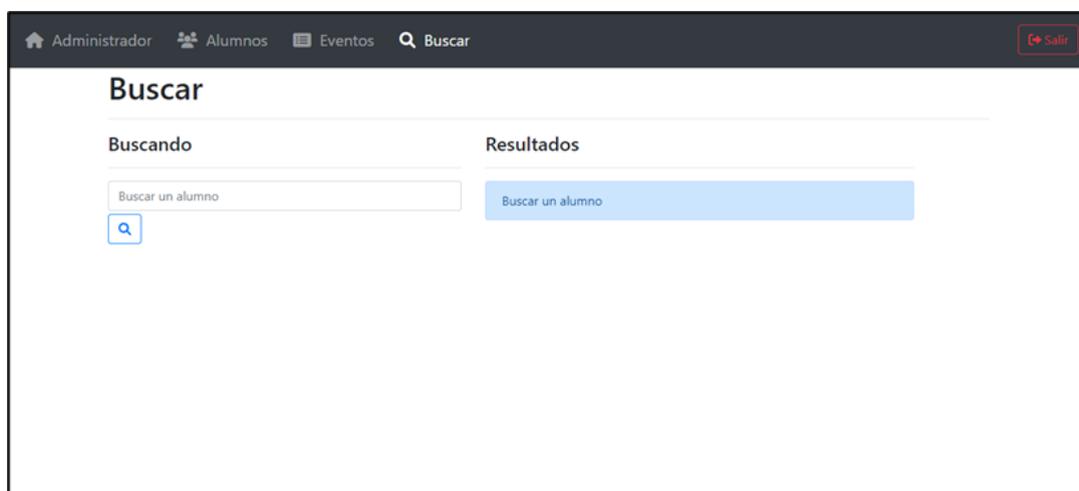
**Ilustración 40** Eliminar un Alumno



**Fuente:** Elaboración propia

La **ilustración 40**, indica la eliminación de un alumno, para poder realizar esta acción al igual que el modificar mencionado en la **ilustración 39**, se debe de seleccionar en la lista de los alumnos el atleta que se requiera borrar, luego se procede con la eliminación del mismo mediante el botón rojo sobre el cual se debe dar click.

**Ilustración 41** Página Buscar Alumno



**Fuente:** Elaboración propia

En la **ilustración 41**, el módulo buscar permite encontrar a un alumno de una manera más rápida, para lograrlo se debe ingresar el nombre o apellido del estudiante y automáticamente se obtendrá su información.

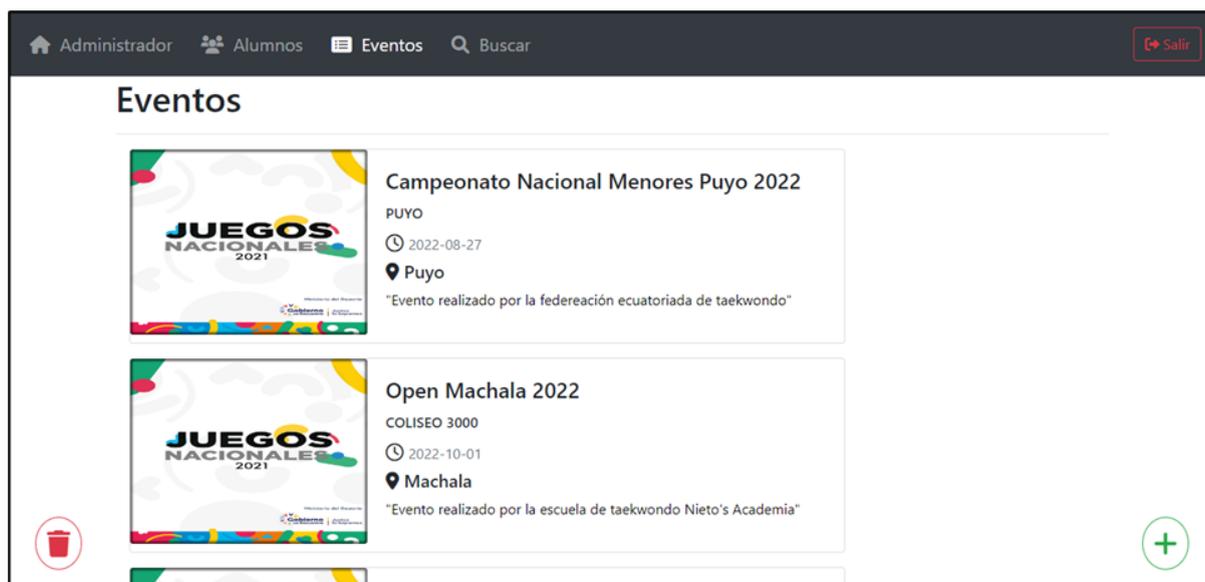
**Ilustración 42** Búsqueda de un alumno



**Fuente:** Elaboración propia

La **ilustración 42**, presenta los resultados de búsqueda realizados en el módulo mencionado en la **ilustración 41**. Como se puede observar el usuario deberá ingresar los datos del estudiante como el nombre o apellido y la aplicación filtrará esa información de tal manera que se obtendrá al alumno esperado.

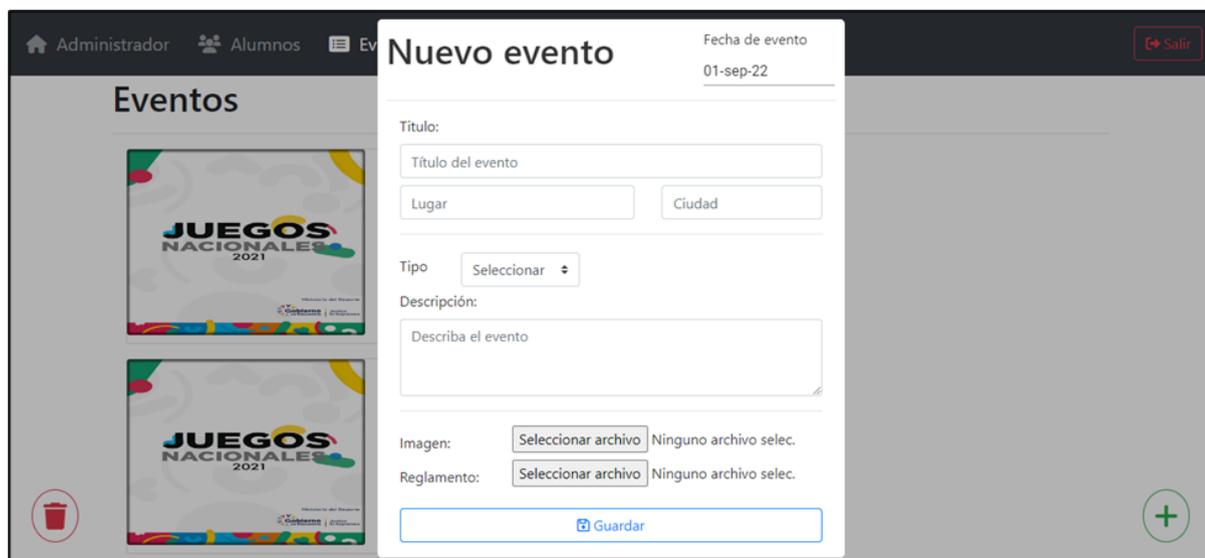
Ilustración 43 Página de Eventos



Fuente: Elaboración propia

La **ilustración 43**, presenta la página de eventos, en la cual se encontrarán todos los campeonatos realizados por la academia y aquellos que serán netamente informativos; en este módulo, el usuario puede informarse sobre los certámenes elaborados por la escuela y las competencias que se realizarán a nivel cantonal, regional y nacional.

Ilustración 44 Formulario para la creación de un evento



Fuente: Elaboración propia

Se puede observar en la **ilustración 44**, el formulario para la creación de un nuevo evento, con información relevante que el administrador debe proporcionar al sistema, es necesario indicar que se cuenta con un apartado denominado

reglamento, el mismo que permite al usuario conocer sobre las normas que se ejecutarán dentro de esa competencia.

### 3. CAPÍTULO III. EVALUACIÓN DEL PROTOTIPO

#### 3.1. Plan de evaluación

El proceso de evaluación de la aplicación desarrollada se basa en la metodología “SWIRL”, la misma que manifiesta tres formas de evaluación que son la verificación de las métricas de calidad, la evaluación de las tendencias de diseño y finalmente por medio de herramientas externas para medir el nivel SEO de la plataforma web.

#### 3.2. Resultados de la evaluación

##### 3.2.1. Evaluación de la calidad de la aplicación web

Los controles de calidad se basan en indicadores especificados según las métricas ISO/IEC 9126 que son usabilidad, confiabilidad, funcionalidad, eficiencia, portabilidad y mantenibilidad [2].

Para la realización de la evaluación de calidad se presentará una matriz con los parámetros considerados para valorar el sistema; debido a que la aplicación no cuenta con un nombre propio se decidió utilizar la plataforma de servicios en la nube Heroku, por lo cual las herramientas web para el análisis de calidad utilizarán la dirección (<https://app-nietos.herokuapp.com/>).

**Tabla 15** Evaluación de calidad ISO/IEC 9126

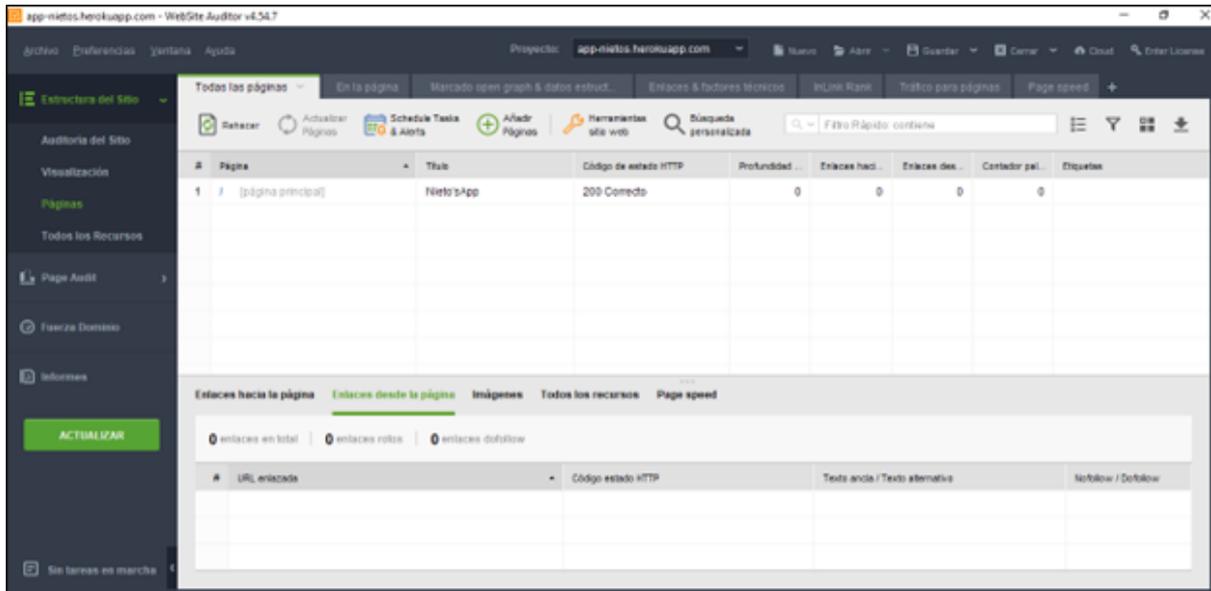
Evaluación del estándar ISO/IEC 9126	Funcionalidad	Fiabilidad	Eficiencia	Mantenibilidad
Website Auditor	x			
GTmetrix		x		
Website Grader			x	x

**Fuente:** Elaboración propia

### 3.2.1.1. Website Auditor

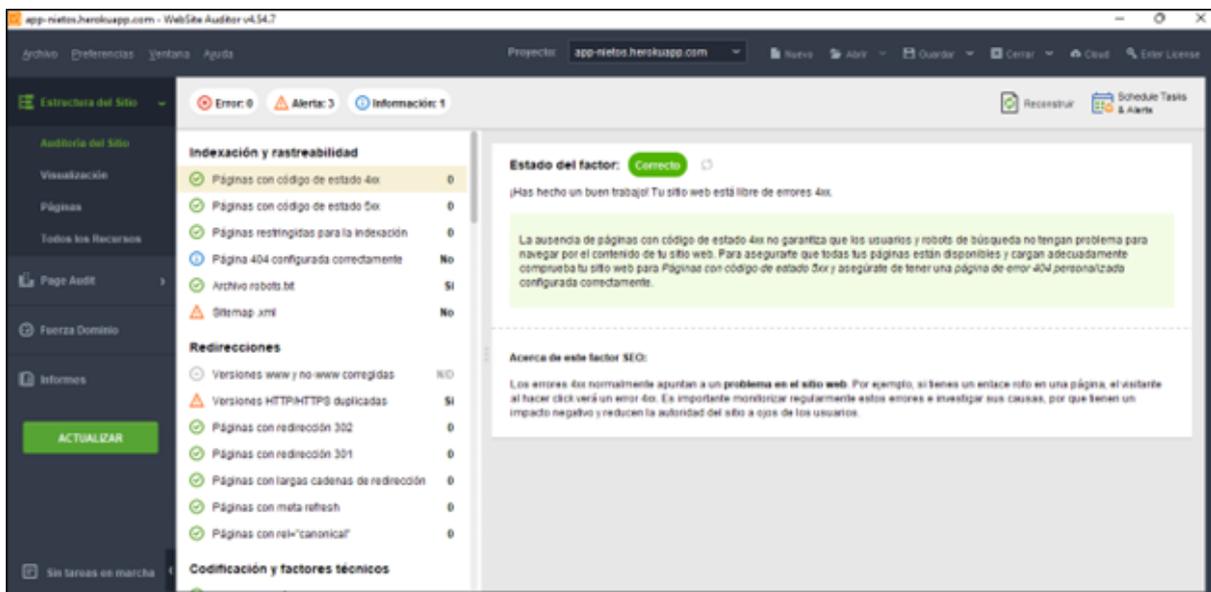
Es una herramienta SEO completa que permite evaluar diferentes características que la norma ISO/IEC 9126 establece como requisito para verificar la calidad de una aplicación web y esos requerimientos son la funcionalidad y la confiabilidad.

Ilustración 45 Evaluación calidad - funcionalidad del sistema



Fuente: Elaboración propia

Ilustración 46 Evaluación calidad - funcionalidad del sistema

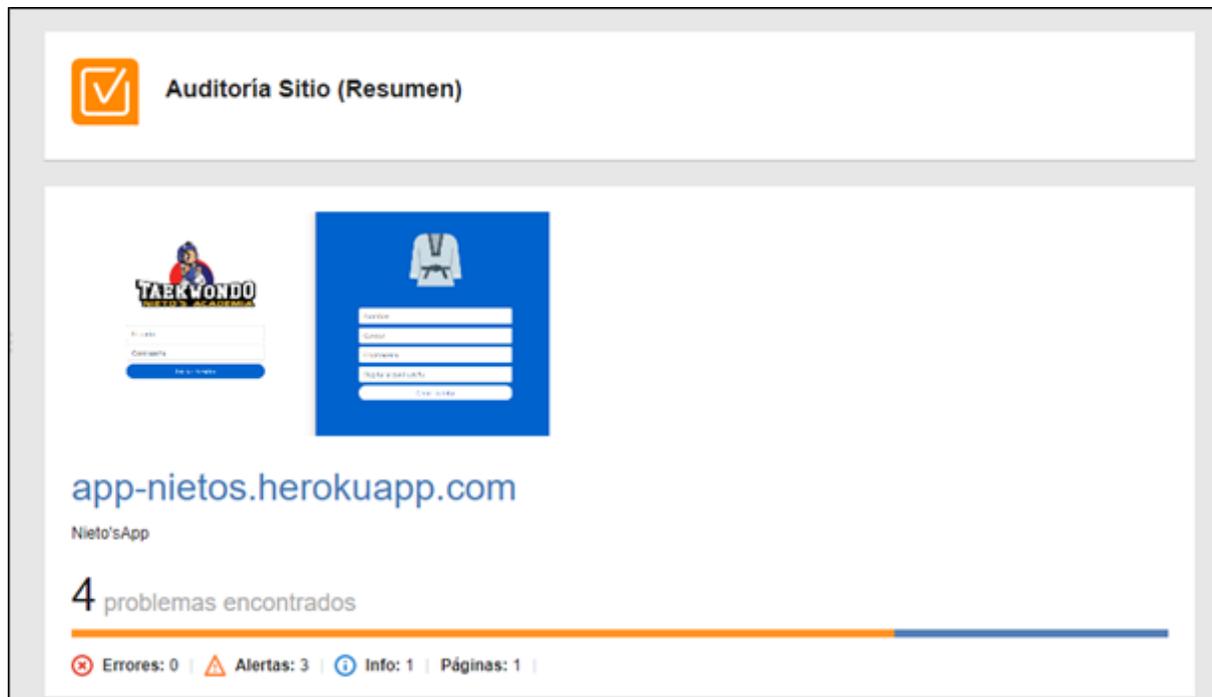


Fuente: Elaboración propia

Como se puede observar en la **ilustración 45 y 46**, la evaluación de funcionalidad del sistema es muy buena, la página del Website Auditor arroja un resultado

correcto lo que indica que el sistema en cuanto a su funcionalidad cumple con los requerimientos necesarios de la norma ISO/IEC 9126.

**Ilustración 47** Evaluación calidad – funcionalidad del sistema



**Fuente:** Elaboración propia

Como se puede observar en la **ilustración 47**, los resultados obtenidos son buenos donde existen 4 problemas que se dividen en 3 alertas y 1 informativa, pero hay que resaltar que estos inconvenientes son más advertencias que provienen de librerías implementadas y estos son problemas de terceros; en conclusión, el sistema cumple con la característica de funcionalidad adecuada comprobando la calidad de la aplicación.

### **3.2.1.2. Website Grader**

Esta herramienta gratuita permite evaluar el rendimiento, la adaptabilidad móvil, el SEO y la seguridad de un sitio web mediante un informe personalizado con los errores y recomendaciones para mejorar la aplicación web.

**Ilustración 48** Evaluación de calidad - eficiencia y mantenibilidad



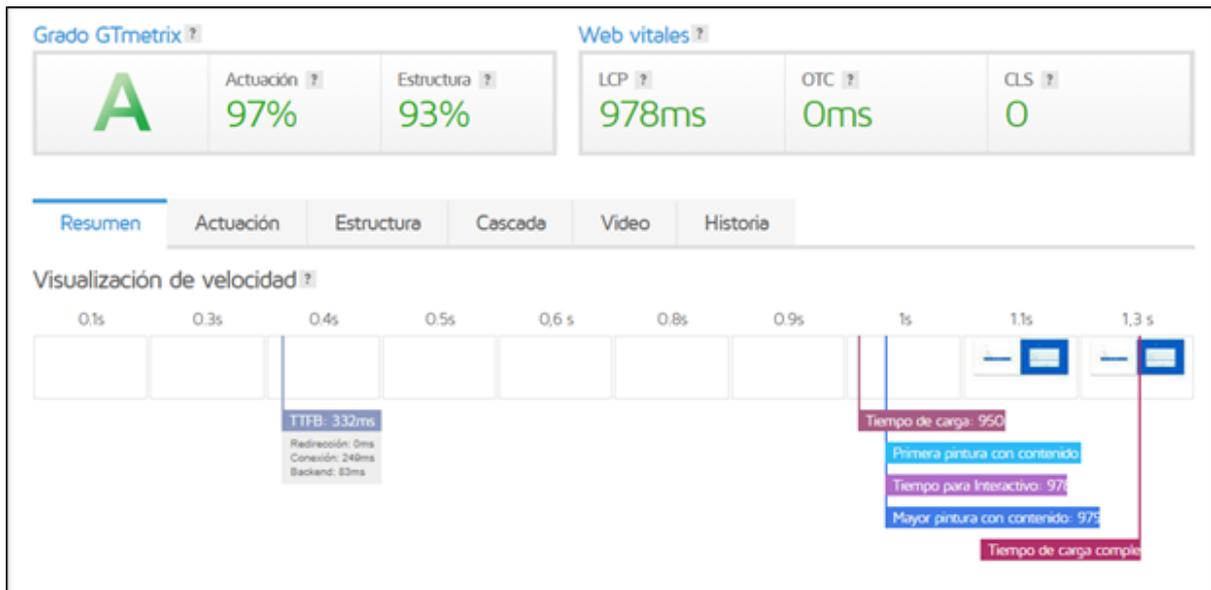
**Fuente:** Elaboración propia

En la **ilustración 48**, se puede observar el resultado que se obtiene de la herramienta “Website Grader”, donde se tiene una calificación 30/30 en cuanto al rendimiento general de la aplicación dividido en tamaño de la página de 743 kb, velocidad de acceso de 5.3 seg, JavaScript minimizado y CSS optimizado; en lo referente al SEO del sistema se tiene un 25/30; una adaptabilidad móvil de 30/30 lo que indica que la plataforma se adapta a cualquier entorno con facilidad mientras que en la seguridad se tiene una calificación de 5/10 lo que indica que se puede mejorar el sitio web.

### **3.2.1.3. GTmetrix**

Esta aplicación determina el grado de fiabilidad del sistema web a través de los tiempos de ejecución entre el navegador y el servidor de aplicación, permitiendo tener una idea clara sobre los tiempos de velocidad de la plataforma.

**Ilustración 49** Evaluación de seguridad - GTmetrix



Fuente: Elaboración propia

En la **ilustración 49**, se puede determinar que los tiempos de ejecución son muy positivos con una evaluación del 97% referente al rendimiento del sistema, un 93% en cuanto a la estructura del mismo y en lo que concierne a la velocidad de carga se tiene un resultado de 978 ms; en conclusión, la herramienta GT-metrix arroja una calificación “A” del funcionamiento general de la aplicación dando a conocer la fiabilidad y confiabilidad de la misma.

Muchas de las herramientas no permiten evaluar índices como la usabilidad y portabilidad de la aplicación web por falta de un dominio privado, por lo tanto, estos parámetros se pueden analizar con base a la observación y experiencia de los usuarios con la aplicación.

La **tabla 17** presentada a continuación indica la forma idónea para la evaluación de la usabilidad y la portabilidad:

**Tabla 16** Evaluación calidad - usabilidad - portabilidad

Evaluación con base a la norma ISO/IEC 9126		Muy malo	Malo	Regular	Bueno	Excelente
Característica	Módulo	1	2	3	4	5
Usabilidad	Aprendizaje					X
	Comprensión				X	
	Operatividad				X	
	Atractividad				X	
Portabilidad	Capacidad de instalación		X			
	Capacidad de reemplazamiento			x		

**Fuente:** Elaboración propia

### 3.3. Conclusiones

- Se desarrolló una aplicación web funcional satisfaciendo la necesidad de automatizar la gestión administrativa de la escuela de taekwondo “Nieto’s Academia”, logrando la implementación de una arquitectura eficiente usando la metodología de desarrollo ágil “SWIRL”.
- La identificación y recolección de requisitos es el proceso principal para lograr establecer la trayectoria y el alcance del proyecto, ello permitirá poder identificar las soluciones y cumplir con lo solicitado por el cliente.
- La implementación de React para el desarrollo del Front-end de la aplicación permitió crear interfaces dinámicas y sencillas de una manera rápida y eficaz reduciendo de manera considerable los tiempos establecidos en la elaboración del mismo y facilitando su interacción con el Back-end.
- La implementación del entorno de ejecución NodeJs en el lado del servidor de la aplicación web permitió crear un sistema rápido, versátil y eficiente capaz de procesar varias peticiones provenientes de diferentes usuarios al mismo tiempo y en conjunto con el gestor de base de datos MongoDB garantizan la integración y seguridad de la información.
- El uso de una metodología de desarrollo ágil como lo es “SWIRL”, facilita la elaboración eficiente y eficaz de los proyectos de software por medio de la implementación de cada una de sus fases, permitiendo realizar aplicaciones de calidad.
- La evaluación de la aplicación web mediante el uso de las diferentes herramientas SEO permitió cumplir con las métricas establecidas en el estándar ISO/IEC 9126 por lo cual se desarrolló un sistema eficiente, óptimo, confiable, adaptable a cualquier entorno y de calidad satisfaciendo las necesidades del cliente.

### **3.4. Recomendaciones**

- Implementar una metodología de desarrollo ágil como “SWIRL” es muy útil para elaborar proyectos de software de calidad, sobre todo cuando existe un tiempo relativamente corto para su ejecución.
- Centrarse en una tecnología de desarrollo tanto para Front-end o Back-end que sea simple, potente y flexible, que permita aprovechar al máximo los recursos con los que se cuente y la facilidad de comunicación e interacción con otras plataformas o servicios.
- Utilizar herramientas como “Bootstrap” reduce el tiempo de elaboración de la parte visual de la aplicación, lo que permite centrarse en la parte operacional del sistema.
- Las bases de datos no relacionales son muy útiles para proyectos que deban manejar gran cantidad de datos y consultas sin afectar la integridad de la información.
- Es muy importante realizar pruebas unitarias al sistema, esto permitirá reducir el mayor número de errores por cada módulo implementado en el sistema.

#### 4. BIBLIOGRAFÍA

- [1] M. E. G. Moyota, F. Imbaquingo, G. Cuascota, M. Quilismal, y J. Jácome, «Uso de sockets en la informática: Una Revisión Sistemática de la literatura: Revisión Sistemática de la literatura: Uso de Sockets en la Informática.», *Innov. Dev. Eng. Appl. Sci.*, vol. 2, n.º 2, Art. n.º 2, 2020, doi: 10.53358/ideas.v2i2.526.
- [2] J. R. M. Ríos y M. de las N. Pedreira-Souto, «SWIRL», *metodología para el diseño y desarrollo de aplicaciones web*. 3Ciencias, 2019.
- [3] P. D. Dutonde, «Website Developmemt Technologies: A Review», *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 10, n.º 1, pp. 359-366, ene. 2022, doi: 10.22214/ijraset.2022.39839.
- [4] D. F. G. Morocho, «Orígenes del taekwondo: un análisis etnográfico de su expansión en Ecuador/Origins of Taekwondo: an ethnographic analysis of its expansion in Ecuador», *PODIUM - Rev. Cienc. Tecnol. En Cult. Física*, vol. 16, n.º 1, Art. n.º 1, feb. 2021.
- [5] R. D. C. Núñez, «LA PRÁCTICA DE LOS FUNDAMENTOS BÁSICOS DEL TAEKWONDO COMO», n.º 65, p. 20, 2020.
- [6] J. Medina Cruz, E. Pineda Ballesteros, F. R. Téllez Acuña, J. Medina Cruz, E. Pineda Ballesteros, y F. R. Téllez Acuña, «Requerimientos de software: prototipado, software heredado y análisis de documentos», *Ing. Desarro.*, vol. 37, n.º 2, pp. 327-345, dic. 2019, doi: 10.14482/inde.37.2.1053.
- [7] V. D. Gil-Vera, J. C. Gil-Vera, C. R. Gomes-Da-Silva, y J. Teutsch, «Frameworks para el desarrollo de prototipos WEB: Un caso de aplicación», *Lámpsakos*, n.º 20, Art. n.º 20, 2018, doi: 10.21501/21454086.2065.
- [8] L. M. N. Magaly, M. R. J. Rolando, M. R. R. Fernando, y P. S. C. Marcela, «Mensajería cliente-servidor aplicando sockets en las herramientas GEANY IDE 1.31, PHYTON 3.7 y POSTGRESQL 9.5 en el sistema operativo CENTOS 7: Client-server messaging applying sockets in GEANY IDE 1.31, PHYTON 3.7 and POSTGRESQL 9.5 tools in CENTOS 7 operating system», *Conf. Proc. Machala*, vol. 4, n.º 1, Art. n.º 1, ago. 2020, doi: 10.48190/cp.v4n1a11.
- [9] M. Peralta-Ascue y J. L. Merma-Aroni, «Integración de procesos de negocio aplicando la arquitectura orientada a servicios (SOA)», *Interfases*, n.º 010, Art. n.º 010, dic. 2017, doi: 10.26439/interfases2017.n10.1771.
- [10] A. Hernández Paez, J. A. Domínguez Falcón, y A. A. Pi Cruz, «Arquitectura de software para el desarrollo de videojuegos sobre el motor de juego Unity 3D», *ID Tecnológico*, vol. 14, n.º 1, pp. 54-65, jun. 2018, doi: 10.33412/idt.v14.1.1803.
- [11] M. J. Blas, H. P. Leone, y S. M. Gonnet, «Modelado y verificación de patrones de diseño de arquitectura de software para entornos de computación en la nube», dic. 2019, doi: 10.17013/risti.35.1-17.
- [12] Z. E. M. Rodríguez, L. D. P. Rodríguez, y J. C. G. Suarez, «Arquitectura basada en Microservicios y DevOps para una ingeniería de software continua», *Ind. Data*, vol. 23, n.º 2, Art. n.º 2, dic. 2020, doi: 10.15381/idata.v23i2.17278.
- [13] M. Á. Niño-Zambrano, G. A. Ramírez-González, J. A. J. Builes, E. G. Pemberty, y A. M. M. Velasco, «La Búsqueda de Interacción Semántica en la Web de las Cosas: Una Revisión», *Investig. E Innov. En Ing.*, vol. 8, n.º 3, Art. n.º 3, nov. 2020, doi: 10.17081/invinno.8.3.4705.
- [14] A. Pilicita Garrido, Y. Borja López, y G. Gutiérrez Constante, «Rendimiento de MariaDB y PostgreSQL», *Rev. Científica Tecnológica UPSE*, vol. 7, n.º 2, pp. 09-16, dic. 2020, doi: 10.26423/rctu.v7i2.538.
- [15] Y. Sadqi y Y. Maleh, «A systematic review and taxonomy of web applications threats», *Inf. Secur. J. Glob. Perspect.*, vol. 31, n.º 1, pp. 1-27, ene. 2022, doi: 10.1080/19393555.2020.1853855.
- [16] M. del P. Paredes Colmenar, *Aplicaciones web*. Madrid: Síntesis, 2020.
- [17] S. M. Velásquez, J. D. V. Montoya, M. E. G. Adasme, E. J. R. Zapata, A. A. Pino, y S. L. Marín, «Una revisión comparativa de la literatura acerca de metodologías

- tradicionales y modernas de desarrollo de software», *Rev. CINTEX*, vol. 24, n.º 2, Art. n.º 2, dic. 2019, doi: 10.33131/24222208.334.
- [18] J. Molina-Ríos y N. Pedreira-Souto, «Comparison of development methodologies in web applications», *Inf. Softw. Technol.*, vol. 119, p. 106238, mar. 2020, doi: 10.1016/j.infsof.2019.106238.
- [19] J. Molina Ríos y N. Pedreira-Souto, «Approach of Agile Methodologies in the Development of Web-Based Software», *Information*, vol. 10, n.º 10, Art. n.º 10, oct. 2019, doi: 10.3390/info10100314.
- [20] N. J. P. Lizcano, D. G. D. Ramírez, D. M. D. Villota, y Y. A. M. Pineda, «Migración del Sistema de Información para la Administración del Talento Humano de la Policía Nacional: una revisión sistemática», *Rev. Logos Cienc. Tecnol.*, vol. 13, n.º 3, Art. n.º 3, dic. 2021, doi: 10.22335/rlct.v13i3.1422.
- [21] «Repositorio Digital de la UTMACH: Página de inicio». <http://repositorio.utmachala.edu.ec/> (accedido 4 de septiembre de 2022).
- [22] S. Kumar, «A REVIEW ON CLIENT-SERVER BASED APPLICATIONS AND RESEARCH OPPORTUNITY», *Int. J. Sci. Res.*, vol. 10, pp. 33857-33862, ago. 2019, doi: 10.24327/ijrsr.2019.1007.3768.
- [23] M. Toscano, M. J. Cobo, y E. Herrera-Viedma, «Software solutions for web information systems in digital humanities: review, analysis and comparative study», *Prof. Inf.*, vol. 31, n.º 2, Art. n.º 2, mar. 2022, doi: 10.3145/epi.2022.mar.11.
- [24] R. Espinosa-Hurtado, «Análisis comparativo para la evaluación de frameworks usados en el desarrollo de aplicaciones web», *CEDAMAZ*, vol. 11, n.º 2, Art. n.º 2, dic. 2021, doi: 10.54753/cedamaz.v11i2.1182.
- [25] «React – Una biblioteca de JavaScript para construir interfaces de usuario». <https://es.reactjs.org/>
- [26] R. Tapia, «React Native: acortando las distancias entre desarrollo y diseño móvil multiplataforma», *Rev. Digit. Univ.*, vol. 20, ago. 2019, doi: 10.22201/codeic.16076079e.2019.v20n5.a5.
- [27] L. A. H. Leyva, S. C. Tamayo, y R. U. Salgado, «Sistema informático para la elaboración y publicación del horario docente de la Universidad de Holguín», *RILCO Rev. Investig. Latinoam. En Compet. Organ.*, n.º 5, p. 5, 2020.
- [28] F. Luna, C. P. Millahual, y M. Iacono, *PROGRAMACION WEB Full Stack 10 - CSS avanzado: Desarrollo frontend y backend - Curso visual y práctico*. RedUsers, 2018.
- [29] M. O. contributors Jacob Thornton, and Bootstrap, «Get started with Bootstrap». <https://getbootstrap.com/docs/5.2/getting-started/introduction/> (accedido 20 de agosto de 2022).
- [30] D. J. Bastidas-Logroño, O. O. Espíndola-Lara, y A. D. Palma-Rivera, «Implementación del bootstrap como una metodología ágil en la web», *Rev. Arbitr. Interdiscip. Koinonía*, vol. 5, n.º 9, p. 268, ene. 2020, doi: 10.35381/r.k.v5i9.648.
- [31] «JavaScript | MDN». <https://developer.mozilla.org/es/docs/Web/JavaScript> (accedido 20 de agosto de 2022).
- [32] K. J. Theisen, «Programming languages in chemistry: a review of HTML5/JavaScript», *J. Cheminformatics*, vol. 11, n.º 1, p. 11, feb. 2019, doi: 10.1186/s13321-019-0331-1.
- [33] N. P. Pérez, «Solución informática para la selección del servidor web durante la migración a código abierto», *Rev. Cuba. Cienc. Informáticas*, vol. 14, n.º 2, pp. 49-69, 2020.
- [34] A. Suárez Zapata, J. Torres País, P. A. Martínez Delgado, R. García Olcina, J. Martos Torres, y J. Soret Medel, *Implementación de un servidor web para gesionar un core time-to-digital-convertir multicanal empleado en sistemas médicos PET*, 1.ª ed. Editorial Científica 3Ciencias, 2018. doi: 10.17993/IngyTec.2018.44.
- [35] I. Odun-Ayo, M. Adebisi, O. Okesola, y O. Vincent, «Cloud middleware and services-a systematic mapping review», *Bull. Electr. Eng. Inform.*, vol. 10, n.º 5, Art. n.º 5, oct. 2021, doi: 10.11591/eei.v10i5.3169.
- [36] M. Bawane, «A Review on Technologies used in MERN stack», *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 10, n.º 1, pp. 479-488, ene. 2022, doi: 10.22214/ijraset.2022.39868.

- [37] C. X. Peraza de Aparicio y N. Y. Zurita Barrios, «Las bases de datos como estrategia didáctica para investigadores noveles», *RECIMUNDO*, vol. 4, n.º 4, pp. 19-29, nov. 2020, doi: 10.26820/recimundo/4.(4).noviembre.2020.19-29.
- [38] P. N. Moya, T. M. Soria, S. V. Arias, y C. C. Guayta, «Análisis de impacto y medición de confiabilidad y tiempo en la migración de bases de datos Sql a Nosql», *Cienc. Digit.*, vol. 2, n.º 3, Art. n.º 3, jul. 2018, doi: 10.33262/cienciadigital.v2i3.138.
- [39] W. K. Hauger y M. S. Olivier, «NOSQL Databases: Forensic Attribution Implications», *SAIEE Afr. Res. J.*, vol. 109, n.º 2, pp. 119-132, jun. 2018, doi: 10.23919/SAIEE.2018.8531952.
- [40] A. Chauhan, «A Review on Various Aspects of MongoDB Databases», *Int. J. Eng. Res. Technol.*, vol. 8, n.º 5, may 2019, doi: 10.17577/IJERTV8IS050031.
- [41] M. Monu y S. Pal, «A Review on Storage and Large-Scale Processing of Data-Sets Using Map Reduce, YARN, SPARK, AVRO, MongoDB», *SSRN Electron. J.*, 2019, doi: 10.2139/ssrn.3365415.
- [42] M. Zhou, «A Review of JavaScript Object Notation in Data Analysis», jun. 2022, pp. 1771-1774. doi: 10.2991/assehr.k.220504.321.
- [43] J. M. Ríos, M. V. Pardo, J. H. Tapia, B. R. Machare, y C. J. Parrales, *Del análisis al diseño*. 3Ciencias, 2019.
- [44] «Software de diagramación y creación de diagramas de flujo | Microsoft Visio». <https://www.microsoft.com/es-ww/microsoft-365/visio/flowchart-software> (accedido 21 de agosto de 2022).
- [45] «Software de administración de proyectos | Microsoft Project». <https://www.microsoft.com/es-ww/microsoft-365/project/project-management-software> (accedido 21 de agosto de 2022).
- [46] «Software de Toad Data Modeler | Las mejores herramientas de diseño de bases de datos». <https://www.quest.com/mx-es/products/toad-data-modeler/> (accedido 21 de agosto de 2022).
- [47] «StarUML». <https://staruml.io/> (accedido 21 de agosto de 2022).
- [48] «Software gratuito de mapeo y modelamiento de procesos de negocio - Bizagi Modeler». <https://www.bizagi.com/es/plataforma/modeler> (accedido 21 de agosto de 2022).
- [49] «Balsamiq Wireframes - Industry Standard Low-Fidelity Wireframing Software | Balsamiq». <https://balsamiq.com/wireframes/> (accedido 21 de agosto de 2022).
- [50] Universidad Nacional Mayor de San Marcos, D. E. Lucero Guevara, N. B. La Serna Palomino, y Universidad Nacional Mayor de San Marcos, «Automatización de requisitos: Historias de usuario generadas a partir de un modelo orientado a objetivos basado en el framework i\*», *Interfases*, n.º 011, pp. 57-72, 2018, doi: 10.26439/interfases2018.n011.2953.
- [51] D. Carrizo, A. Alfaro, D. Carrizo, y A. Alfaro, «Método de aseguramiento de la calidad en una metodología de desarrollo de software: un enfoque práctico», *Ingeniare Rev. Chil. Ing.*, vol. 26, n.º 1, pp. 114-129, mar. 2018, doi: 10.4067/S0718-33052018000100114.
- [52] J. E. G. Burgués, *Aprende a Modelar Aplicaciones con UML- Tercera Edición*. IT Campus Academy, 2018.
- [53] M. L. Roldán, M. Vegetti, M. Marciszack, S. Gonnet, y H. Leone, «Un Modelo Conceptual para la Especificación y Trazabilidad de Requerimientos Funcionales basados en Casos de Uso y Casos de Prueba», p. 12, 2017.

## 5. ANEXOS

### ANEXO A: ANÁLISIS DE RIESGOS

Tabla 17 Gestión de Riesgos

N°	RIESGO	DETONANTE	PREVENCIÓN
RG-1	Recolección incompleta de requisitos.	No dialogar con los interesados del proyecto.	Mantener un diálogo continuo vía conferencia.
RG-2	Falta de conocimiento en la implementación de nuevas tecnologías.	Falta de documentación y guía de desarrollo.	Investigar herramientas actuales con guías de soporte
RG-3	Falta de recursos tecnológicos para la implementación de la aplicación.	No definir el alcance del proyecto a nivel de arquitectura.	Utilizar alternativas de implementación del sistema.
RG-4	Daño de los equipos físicos de la aplicación.	Catástrofe natural	Hacer respaldos de la codificación vía GitHub.
RG-5	Incompatibilidad de herramientas de desarrollo.	Malas prácticas en el desarrollo del sistema.	Verificar la guía de compatibilidad y documentación de herramientas.
RG-6	Los procesos de diseño y desarrollo no finalizan en el tiempo estimado.	Disponibilidad de tiempo.	Seguir el cronograma de actividades estableciendo holuras.

Fuente: Elaboración propia

## ANEXO B: FICHA DE INSCRIPCIÓN DE LA ESCUELA "NIETO'S ACADEMIA"

Ilustración 50 Ficha de inscripción "Nieto's Academia"



[Empty rectangular box for photo or signature]

### Ficha de Inscripción

Fecha de inscripción

Apellidos				
Nombres				
Fecha de Nacimiento				
Número de Cédula				
Peso				
Estatura				
Sexo	F		M	
Somato tipo				
Patología				
Nombre del representante				
Número de teléfono				
Dirección domiciliaria				
Institución Educativa				
Observación				

Fuente: Elaboración propia