



PLATAFORMA WEB INTELIGENTE PARA LA OPTIMIZACIÓN Y AUTOMATIZACIÓN DE LA GESTIÓN DE EVENTOS DEPORTIVOS DEPORTIVOS

Documento de requerimientos de software

29 de octubre de 2024



Kevin Tapia
Jhonyl Sucuy



Tabla de contenido

Historial de Versiones	1
Información del Proyecto	2
Aprobaciones	2
1. Propósito	2
2. Alcance del producto / Software	3
3. Referencias	3
4. Funcionalidades del producto	3
5. Clases y características de usuarios	4
6. Entorno operativo	6
7. Requerimientos funcionales	6
8. Reglas de negocio	9
9. Requerimientos de interfaces externas	15
9.1. Interfaces de usuario	¡Error! Marcador no definido.
9.2. Interfaces de hardware	16
9.3. Interfaces de software	16
9.4. Interfaces de comunicación	16
10. Requerimientos no funcionales	17
11. Otros requerimientos	17
12. Glosario	17

Historial de Versiones

Fecha	Versión	Autor	Organización	Descripción

--	--	--	--	--

Información del Proyecto

Empresa / Organización	Federación Deportiva Provincial Estudiantil de Napo
Proyecto	
Fecha de preparación	
Cliente	
Patrocinador principal	
Gerente / Líder de Proyecto	Kevin Tapia
Gerente / Líder de Análisis de negocio y requerimientos	

Aprobaciones

Nombre y Apellido	Cargo	Departamento u Organización	Fecha	Firma

1. Propósito

Este documento define las especificaciones para el software denominado Athly, cuya versión de lanzamiento prevista es v1.0. Athly será una plataforma web inteligente destinada a la optimización y automatización de la gestión de eventos deportivos para la Federación Deportiva Provincial de Napo.

El alcance de este documento cubre la totalidad del sistema que se planea desarrollar en la versión v1.0. Incluye los componentes clave, como los módulos de autenticación y manejo de roles, los sistemas de registro de jugadores, generación de horarios utilizando algoritmos de optimización, selección de estudiantes destacados mediante inteligencia artificial, gestión de notificaciones, pagos en línea y generación automática de carnets. Se espera que el desarrollo de estos componentes se realice de manera integral para cumplir con los objetivos definidos del proyecto.

2. Alcance del producto / Software

1. Propósito general:

El objetivo principal de la plataforma destinada a la federación es optimizar la gestión de los eventos deportivos anuales que organiza. Esto se logrará mediante la mejora en la recolección de datos, la optimización de la comunicación con los participantes en el proceso, la reducción de los tiempos asociados a la gestión y la generación de procedimientos necesarios para la realización de dichos eventos, así como la facilitación del acceso a los resultados obtenidos.

2. Beneficios que brinda al área de negocio:

Al centralizar todas las operaciones relacionadas, se reducirá el tiempo y los recursos necesarios para la administración manual, así como el tiempo requerido para llevar a cabo los eventos. Esto permitirá a la federación enfocarse en otros procesos importantes, fuera del flujo de trabajo habitual. Asimismo, no solo los encargados se beneficiarán del sistema, sino también las partes involucradas en los procesos, ya que no tendrán que realizar trámites presenciales para la entrega o solicitud de documentos importantes; de este modo, se fomentará la transparencia del proceso, dado que la gestión de resultados será pública y no se limitará a un procedimiento interno de la federación.

3. Objetivos y metas:

- Incrementar la eficiencia operativa y reducir costos administrativos.
- Mejorar el registro y la organización de recursos.
- Facilitar la comunicación entre participantes y organizadores.
- Generar calendarios de juegos de manera eficiente.
- Implementar análisis de rendimiento basado en inteligencia artificial.
- Aumentar la transparencia en la gestión de resultados deportivos.

3. Referencias

Perfil y alcance: [Deber n3 - Perfil y alcance del proyecto a desarrollar V2.pdf](#)

Autores: Jhonyl Sucuy, Kevin Tapia

4. Funcionalidades del producto

1. Gestión de usuarios y permisos de acceso

2. Carga de reglamento y programación del congresillo
3. Sistema de notificaciones por correo e internas
4. Inscripción de jugadores por institución
5. Generación de horarios
6. Selección de estudiantes destacados
7. Registro de datos de partidos
8. Generación de carnets de para los deportistas
9. Realización de pagos por parte de las instituciones educativas

5. Clases y características de usuarios

1. Administrador del sistema

- *Características:*

Tiene el más alto nivel de privilegios dentro de la plataforma y es responsable de gestionar y configurar todos los aspectos del sistema.

- *Funciones relevantes:*

- Agregación, gestión de roles y permisos de usuario.
- Administración de contenido, como la carga de reglamentos y programación de reuniones.
- Configuración de notificaciones y seguimiento de su envío.
- Monitoreo y administración de registros y sanciones a instituciones educativas.
- Gestión de métodos de pago y revisión del historial de transacciones.

- *Frecuencia de uso:*

Alta. Utiliza la mayor parte de las funcionalidades del sistema de forma frecuente para mantener la operatividad de la plataforma.

2. Representante de institución

- *Características:*

Usuarios que representan a cada colegio o institución educativa. Tienen permisos específicos para inscribir jugadores y gestionar la información de su institución.

- *Funciones relevantes:*

- Registro de jugadores por deporte y categoría.
- Visualización y descarga de reglamentos y programación de reuniones.
- Consulta de sanciones y pagos en línea.

- Visualización y descarga de carnets de los deportistas.
- *Frecuencia de uso:*
Moderada. Acceden al sistema principalmente durante las fases de inscripción y cuando se requiere consultar información administrativa.

3. Secretaria

- *Características:*
Usuario encargado de validar las inscripciones y mantener actualizada la base de datos del sistema.
- *Funciones relevantes:*
 - Revisión y validación de la información de los jugadores inscritos.
 - Gestión de sanciones y generación de carnets.
 - Supervisión de los registros y verificación de la asistencia en reuniones.
- *Frecuencia de uso:*
Alta. Utiliza el sistema con frecuencia para mantener la información y documentos actualizados.

4. Árbitro

- *Características:*
Usuario responsable de registrar los resultados y estadísticas de los partidos.
- *Funciones Relevantes:*
 - Registro de resultados y estadísticas de los partidos.
 - Edición y corrección de los datos del partido con validaciones adecuadas.
- *Frecuencia de Uso:*
Moderada. Utiliza el sistema principalmente durante los eventos deportivos y para la gestión de resultados.

5. Usuarios visitantes en el sistema

- *Características:*
Usuarios que participan en los eventos deportivos y que pueden consultar sus resultados y rendimiento.
- *Funciones Relevantes:*
 - Visualización de su información y resultados de competencias.
 - Consulta de horarios y reglamentos.
- *Frecuencia de Uso:*

Baja. Acceden al sistema de forma esporádica para consultar información específica sobre eventos.

6. Entorno operativo

- La plataforma debe ser accesible desde dispositivos de escritorio, portátiles, tablets y smartphones. Los dispositivos cliente deben contar con al menos 2 GB de RAM y un procesador de doble núcleo para asegurar un rendimiento adecuado.
- Los sistemas operativos compatibles incluyen Windows 10 o superior, macOS 10.13 (High Sierra) o superior, Android 8.0 o superior, e iOS 12 o superior, en dispositivos móviles y tablets.

7. Requerimientos funcionales

7.1. Gestión de usuarios y permisos de acceso

- *Descripción:*

Permitir la creación y administración de cuentas para los diferentes tipos de usuario (administrador, representante de colegio, árbitro e invitado), con permisos de acceso específicos.

- *Prioridad:* Alta

- *Acciones iniciadoras y comportamiento esperado:*

- El administrador del sistema accede al módulo de gestión de usuarios.
- El administrador agrega una lista con correos electrónicos, nombres de cada usuario y rol asignado.
- El sistema envía automáticamente un correo a cada usuario con una contraseña temporal y un enlace de activación de cuenta.
- El usuario, al ingresar por primera vez, es dirigido a una pantalla donde se le solicita cambiar su contraseña.
- El usuario establece una nueva contraseña, y el sistema confirma el cambio exitoso.

- *Requerimientos funcionales:*

- **REQ-1:** El sistema debe permitir al administrador agregar el nombre y correo electrónico de cada usuario nuevo y asignarle un rol.
 - **Comportamiento esperado:** El sistema presentará un formulario que incluirá los campos obligatorios: nombre, correo electrónico y rol. Una vez que se complete la información, se habilitará un botón para agregar el usuario. Al agregar el usuario, se mostrará una lista que incluirá el nuevo usuario, así como opciones para modificar y eliminar registros de la lista. Al final, se dispondrá de un botón para agregar usuarios desde la lista. Al

hacer clic en este botón, se mostrará un mensaje de confirmación que permitirá aceptar o cancelar la acción.

- **Condiciones de error:** Si el correo electrónico ingresado no es válido, el sistema deberá mostrar un mensaje de error y no permitirá la creación del usuario hasta que se corrija el campo correspondiente. De igual manera, se aplicará esta restricción al formato del nombre ingresado, asegurando que ambos campos cumplan con los criterios de validación establecidos antes de proceder con la creación del usuario.
- **REQ-2:** El sistema debe enviar un correo electrónico de bienvenida a cada usuario nuevo, incluyendo una contraseña temporal y un enlace para acceder al sistema.
 - **Comportamiento esperado:** El sistema debe mostrar una notificación que confirme el envío de los correos electrónicos. Posteriormente, redirigirá al administrador a una pantalla que exhiba la lista de usuarios registrados en el sistema, incluyendo todos sus datos, el rol asignado y las acciones disponibles, como editar o eliminar a cada usuario del sistema.
- **REQ-3:** Al ingresar por primera vez, el usuario debe ser redirigido automáticamente a una pantalla de cambio de contraseña.

7.2. Carga de reglamento y programación de congresillo

- *Descripción:*
Permitir la carga de reglamentos y la programación de reuniones.
- *Prioridad:* Alta
- *Acciones iniciadoras y comportamiento esperado:*
 - Los administradores cargan reglamentos y definen las reuniones.
 - El sistema envía recordatorios automáticos.
- *Requerimientos Funcionales:*
 - **REQ-1:** El sistema debe permitir la carga de reglamentos en formato PDF por parte del administrador.
 - **Comportamiento esperado:** Al hacer clic en "Cargar Reglamento", se abrirá un formulario que incluirá la opción de seleccionar y cargar un archivo PDF.
 - **Condiciones de error:** Si el archivo subido no es un PDF, el sistema mostrará un mensaje de error indicando el formato incorrecto.
 - **REQ-2:** El sistema debe permitir la programación de reuniones (congresillos) ingresando una fecha, hora y enlace.
 - **Comportamiento esperado:** El administrador llenará un formulario con campos de fecha, hora y enlace de la reunión, y podrá guardar o cancelar.

- **Condiciones de error:** Si se introduce una fecha u hora inválida, el sistema mostrará un mensaje de advertencia y solicitará correcciones.
- **REQ-3:** El sistema debe enviar automáticamente recordatorios de reuniones a los participantes.
 - **Comportamiento esperado:** Los usuarios reciben notificaciones con los detalles de la reunión.
 - **Condiciones de error:** Notificación al administrador si los correos de recordatorio fallan.

7.3. Sistema de notificaciones por correo e internas

- *Descripción:*
Enviar notificaciones por correo electrónico e internas a los usuarios.
- *Prioridad:* Media
- *Acciones iniciadoras y comportamiento esperado:*
 - El sistema dispara notificaciones ante eventos o acciones predefinidas.
- *Requerimientos funcionales:*
 - **REQ-1:** El sistema debe enviar correos electrónicos automáticos a los usuarios para notificar eventos importantes.
 - **Comportamiento esperado:** Generación de un correo basado en una plantilla predefinida.
 - **Condiciones de error:** Mensaje al administrador si el envío de correos falla.
 - **REQ-2:** El sistema debe permitir notificaciones internas visibles en la plataforma.
 - **Comportamiento esperado:** Los usuarios verán una lista de notificaciones en su panel principal.
 - **Condiciones de error:** Notificación al administrador si las notificaciones internas no se pueden mostrar.

7.4. Inscripción de jugadores por institución

- *Descripción:*
Permitir la inscripción de jugadores por parte de los representantes de instituciones.
- *Prioridad:* Alta
- *Acciones Iniciadoras y Comportamiento Esperado:*

- Los representantes ingresan y cargan la información de los jugadores.
- La secretaria valida y aprueba las inscripciones.
- *Requerimientos Funcionales:*
 - **REQ-1:** El sistema debe permitir a los representantes cargar documentos en formato PDF para cada jugador inscrito.
 - Comportamiento esperado: El sistema mostrará un formulario que incluirá la opción de cargar documentos mediante un botón de “Subir archivo”.
 - Condiciones de error: Si el archivo cargado no es un PDF o excede el tamaño máximo permitido, se mostrará un mensaje de error solicitando un archivo válido.
 - **REQ-2:** El sistema debe permitir a los representantes ingresar los datos de los jugadores.
 - Comportamiento esperado: El sistema mostrará un formulario con campos obligatorios.
 - Condiciones de error: Mensaje de error si uno de los campos obligatorios no se completa o contiene datos inválidos (por ejemplo, edad fuera de un rango permitido).
 - **REQ-3:** El sistema debe notificar automáticamente a la Secretaría cuando un representante completa la inscripción y carga de documentos.
 - Comportamiento esperado: La Secretaría recibe una notificación en el sistema y por correo electrónico, indicando que hay inscripciones pendientes de validación.
 - Condiciones de error: Notificación al administrador si la notificación a la Secretaría no se puede enviar.
 - **REQ-4:** El sistema debe permitir a la Secretaría revisar la información y los documentos cargados por cada jugador antes de aprobar la inscripción.
 - Comportamiento esperado: La Secretaría verá una lista de jugadores pendientes de validación, con opciones para visualizar documentos PDF y aprobar o rechazar cada inscripción.
 - Condiciones de error: Mensaje de error si hay un problema al visualizar los documentos o validar los datos.
 - **REQ-5:** El sistema debe registrar el estado de cada inscripción como “Pendiente”, “Aprobada” o “Rechazada” y permitir a los representantes visualizar dicho estado.
 - Comportamiento esperado: Una vez que la Secretaría valida la información, el estado se actualiza automáticamente y los representantes pueden ver el estado en sus perfiles.
 - Condiciones de error: Mensaje de error si no se puede actualizar o visualizar el estado de una inscripción.

7.5. Generación de horarios

- *Descripción:*
Generar automáticamente horarios para los eventos deportivos.
- *Prioridad:* Alta
- *Acciones iniciadoras y comportamiento esperado:*
 - Los administradores ingresan los parámetros y restricciones.
 - El sistema genera y muestra los horarios
- *Requerimientos funcionales:*
 - **REQ-1:** El sistema debe permitir ingresar parámetros como número de equipos, disponibilidad de espacios deportivos y restricciones.
 - **Comportamiento esperado:** Formulario de entrada de datos antes de ejecutar la generación de horarios.
 - **Condiciones de error:** Mensaje de error si los parámetros están incompletos.
 - **REQ-2:** El sistema debe generar automáticamente un horario de juegos.
 - **Comportamiento esperado:** Generación del horario y presentación de vista previa al presidente.
 - **Condiciones de error:** Mensaje de ajuste si el sistema no puede generar un horario viable.
 - **REQ-3:** El sistema debe permitir exportar el horario generado en formato PDF o Excel.
 - **Comportamiento esperado:** Opción para descargar el archivo desde la vista previa.
 - **Condiciones de error:** Notificación al administrador si el archivo no se puede generar.

7.6. Selección de estudiantes destacados

- *Descripción:*
Identificación de estudiantes con buen rendimiento en competencias deportivas.
- *Prioridad:* Alta
- *Acciones iniciadoras y comportamiento esperado:*
El administrador inicia el proceso para analizar el rendimiento de los estudiantes. La inteligencia artificial selecciona a los estudiantes destacados basándose en los datos registrados

- *Requerimientos funcionales:*
 - **REQ-1:** El sistema debe permitir que el administrador inicie el análisis de datos para identificar a los estudiantes destacados.
 - **Comportamiento esperado:** El administrador tendrá acceso a un botón o sección para comenzar la evaluación de datos. El sistema debe presentar una opción para elegir el evento o deporte que se desea analizar.
 - **Condiciones de error:** Si no hay datos suficientes registrados para un evento, el sistema deberá notificarlo con un mensaje claro y permitir al administrador completar la información faltante antes de iniciar el análisis.
 - **REQ-2:** El sistema debe generar un informe de los estudiantes destacados con sus respectivos logros y calificaciones.
 - **Comportamiento esperado:** Después de realizar el análisis, se mostrará un listado con los estudiantes seleccionados, resaltando sus habilidades y logros. El sistema debe ofrecer una opción para exportar este informe a PDF o Excel.
 - **Condiciones de error:** Si los datos de un estudiante presentan inconsistencias, el sistema deberá marcar al estudiante como "revisar" y no incluirlo en el informe hasta que el administrador corrija los datos.
 - **REQ-3:** El sistema debe permitir a los administradores y representantes consultar los resultados y criterios utilizados en la selección.
 - **Comportamiento esperado:** Los administradores y representantes podrán acceder a una sección de "Resultados de selección" donde se expliquen los criterios aplicados para seleccionar a los estudiantes destacados, y se les permita descargar el informe.
 - **Condiciones de error:** Si un representante o administrador no tiene permisos adecuados para ver los resultados, el sistema deberá mostrar un mensaje de acceso denegado.

7.7. Registro de Datos del Partido

- *Descripción:*
Registrar datos relevantes de los partidos, incluyendo resultados y estadísticas.
- *Prioridad:* Media
- *Acciones iniciadoras y comportamiento esperado:*
 - Los árbitros registran los resultados de los partidos.
 - Los administradores gestionan información registrada.
- *Requerimientos funcionales:*

- **REQ-1:** El sistema debe permitir al árbitro ingresar los datos de los partidos (resultados, estadísticas, comentarios).
 - **Comportamiento esperado:** Un formulario con campos específicos y un botón de "Guardar".
 - **Condiciones de error:** Si los campos requeridos están vacíos, se muestra un mensaje de error.
- **REQ-2:** El sistema debe permitir la edición de los datos de los partidos, registrando versiones de cambios.
 - **Comportamiento esperado:** Los administradores podrán editar los datos siempre y cuando se notifique a los equipos involucrados y estos aprueben.
 - **Condiciones de error:** Si la edición falla, se notificará al administrador para que revise.

7.8. Generación de Carnets para los deportistas

- *Descripción:*
Crear y generar carnets automáticamente para los deportistas.
- *Prioridad:* Media
- *Acciones iniciadoras y comportamiento esperado:*
 - Los administradores aprueban las inscripciones.
 - El sistema genera y permite la descarga de carnets.
- *Requerimientos funcionales:*
 - **REQ-1:** El sistema debe permitir la generación automática de carnets una vez se aprueben las inscripciones.
 - **Comportamiento esperado:** Los carnets se generarán y se podrán descargar en formato PDF
 - **Condiciones de error:** Si la generación falla, se mostrará un mensaje de error indicando la causa.

7.9. Realización de Pagos por Parte de las Instituciones Educativas

- *Descripción:*
Permitir a las instituciones educativas realizar pagos de sanciones través de la plataforma.
- *Prioridad:* Alta
- *Acciones iniciadoras y comportamiento esperado:*
El representante de la institución accede a la sección de pagos, selecciona la transacción correspondiente y procede a efectuar el pago de manera segura y directa desde la plataforma.

- *Requerimientos funcionales:*

- **REQ-1:** El sistema debe permitir que las instituciones visualicen las sanciones pendientes.
 - **Comportamiento esperado:** Al acceder a la sección de pagos, el representante de la institución verá una lista de las sanciones, con detalles como el concepto, monto y fecha de vencimiento.
 - **Condiciones de error:** Si no existen pagos pendientes, el sistema deberá mostrar un mensaje que indique "No hay pagos pendientes".
- **REQ-2:** El sistema debe ofrecer la opción de realizar pagos mediante diversos métodos de pago en línea.
 - **Comportamiento esperado:** Al seleccionar un pago pendiente, el sistema presentará diferentes opciones como tarjeta de crédito, débito, y servicios de pago electrónicos. Después de elegir un método, se mostrará un formulario para ingresar los datos del pago de manera segura.
 - **Condiciones de error:** Si los datos ingresados para el pago son incorrectos o no cumplen con los requisitos de seguridad, el sistema deberá mostrar un mensaje de error y ofrecer la posibilidad de corregir la información antes de intentar el pago nuevamente.
- **REQ-3:** El sistema debe generar un comprobante de pago para la institución educativa.
 - **Comportamiento esperado:** Al completar un pago exitoso, se generará automáticamente un comprobante que incluya el detalle de la transacción y se mostrará un mensaje de confirmación. Además, el sistema deberá ofrecer la opción de descargar el comprobante en formato PDF o enviarlo por correo electrónico.
 - **Condiciones de error:** Si la transacción no se puede completar debido a un error en el sistema o en la conexión, se deberá mostrar un mensaje de advertencia, y el pago se marcará como "pendiente de revisión" hasta que se resuelva el problema.

8. Reglas de negocio

1. Acceso y Roles de Usuario:

- Solo los administradores pueden agregar, modificar o eliminar usuarios, así como asignar roles y gestionar permisos de acceso.
- Los usuarios deben iniciar sesión utilizando sus credenciales y, dependiendo de su rol, podrán acceder únicamente a las funcionalidades y secciones correspondientes.
- Los administradores son los únicos que pueden cambiar permisos de otros usuarios.

2. Carga de Reglamentos y Organización de Reuniones:

- Solo los administradores tienen la autorización para cargar reglamentos, definir enlaces de reuniones y enviar recordatorios automáticos a los participantes.

- La asistencia a las reuniones debe registrarse automáticamente al momento de ingresar al enlace, y solo los participantes autorizados pueden unirse a las reuniones programadas.

3. Notificaciones:

- Los administradores pueden configurar qué tipo de notificaciones se enviarán y a quién, basadas en eventos específicos del sistema (nuevos registros, próximas reuniones, pagos pendientes, etc.).
- Los usuarios recibirán notificaciones por correo electrónico e internamente en la plataforma, dependiendo de su rol y configuraciones de notificación.

4. Inscripción de Jugadores:

- Solo los representantes de las instituciones educativas tienen permiso para cargar información y documentos de los jugadores.
- La inscripción de jugadores debe ser revisada y validada por la secretaría antes de ser aprobada. Solo los usuarios con permisos de secretaría pueden aprobar o rechazar inscripciones.

5. Generación de Horarios:

- Los horarios deben ser generados únicamente por los administradores utilizando modelos de optimización que respeten las reglas y parámetros previamente establecidos.
- Solo los administradores tienen acceso para modificar los horarios una vez generados, y cualquier cambio debe ser notificado automáticamente a los colegios, estudiantes y árbitros afectados.

6. Selección de Estudiantes Destacados:

- La selección de estudiantes destacados debe ser realizada por el sistema de inteligencia artificial basado en criterios establecidos previamente por los administradores.
- Los representantes y estudiantes pueden acceder a sus resultados y evaluaciones, pero no pueden modificarlos.

7. Registro y Modificación de Resultados de Partidos:

- Solo los árbitros designados tienen permisos para ingresar los resultados de los partidos.
- Los administradores pueden revisar y, de ser necesario, realizar correcciones en los datos de los partidos, siempre y cuando se notifique a las instituciones involucradas y estas validen, manteniendo un registro de versiones para garantizar la integridad.

8. Generación de Carnets de Deportistas:

- Solo los administradores pueden aprobar la generación de carnets una vez que las inscripciones sean validadas.
- Los representantes tienen permisos para descargar los carnets generados de sus jugadores.

9. Realización de Pagos:

- Las instituciones educativas solo pueden realizar pagos de sanciones o tarifas pendientes que hayan sido previamente registradas y aprobadas por los administradores.
- Todas las transacciones deben ser validadas antes de ser procesadas y se deben generar comprobantes de pago automáticamente al finalizar la transacción.
- Para hacer cumplir las reglas de negocio, podría ser necesaria la definición de requerimientos funcionales que aplican a todo el sistema, no a una funcionalidad específica.

9. Requerimientos de interfaces externas

9.1. Interfaces de usuario

1. Organización de pantalla:

- *Encabezado*: Todas las pantallas deben incluir un encabezado con el logotipo de la federación y un menú de navegación para las áreas disponibles según el rol del usuario.
- *Panel lateral o barra de navegación*: Una barra de navegación lateral para los módulos principales del sistema, con accesos directos y visualización clara de las secciones.
- *Contenido principal*: Área central donde se muestra la información relevante o los formularios de cada módulo.
- *Pie de página*: Incluir información de contacto y enlaces rápidos (términos y condiciones, política de privacidad).

2. Estándares de botones y controles:

- *Estilo de botones*: Todos los botones deben tener un diseño consistente, con colores que indiquen su acción (por ejemplo, verde para confirmar, rojo para cancelar).
- *Botón de acción primaria*: En cada pantalla, debe haber un botón de acción principal que esté destacado (ej., "Guardar" o "Enviar").
- *Estandarización de nombres de botones*: Los nombres de botones deben ser consistentes en todas las pantallas, como "Agregar Usuario", "Editar", "Eliminar", "Guardar Cambios".
- *Indicadores de carga y retroalimentación*: Cuando una acción está en proceso, mostrar indicadores de carga y mensajes de éxito o error para informar al usuario.

9.2. Interfaces de hardware

1. Dispositivos soportados:

El sistema es totalmente compatible con navegadores web modernos en sistemas operativos como Windows, macOS y Linux. De igual forma, optimizado para acceso desde smartphones y tabletas.

2. Protocolos de comunicación soportados:

HTTP/HTTPS para transferir datos entre el sistema y los navegadores en dispositivos de escritorio y móviles.

3. Interacción de datos y control entre software y hardware:

Los administradores y encargados pueden generar carnets para estudiantes y otros documentos oficiales. El sistema debe permitir la descarga de documentos en formato PDF, compatible con cualquier impresora conectada al dispositivo, o bien con impresoras de tarjetas para imprimir directamente.

9.3. Interfaces de software

1. Base de datos:

El sistema utiliza una base de datos relacional para gestionar los datos de usuarios, eventos deportivos, inscripciones de estudiantes, resultados y demás registros.

2. Librerías y Frameworks

El sistema utiliza FastApi como framework backend, proporcionando una estructura robusta y modular para gestionar la lógica del servidor y las solicitudes HTTP. En el frontend, se utiliza React para desarrollar una interfaz de usuario interactiva y dinámica.

9.4. Interfaces de comunicación

- Todo el tráfico de datos entre el cliente (usuario) y el servidor se realiza a través del protocolo HTTPS, que proporciona encriptación y seguridad mediante SSL/TLS.
- El sistema utiliza un servicio de correo, como un servidor SMTP configurado o la API de SendGrid, para enviar notificaciones automáticas, contraseñas temporales
- Los correos que incluyen enlaces sensibles, como enlaces para restablecer contraseñas, deben estar protegidos mediante tokens temporales de un solo uso y válidos por un tiempo limitado

- Todos los formularios electrónicos tendrán validaciones tanto en el frontend como en el backend.

10. Requerimientos no funcionales

1. **NF-1:** Tiempo de Respuesta: El sistema debe responder a cada solicitud del usuario en un tiempo menor a 3 segundos para las acciones de consulta y carga de páginas, con un tiempo máximo de 5 segundos para operaciones más complejas
2. **NF-2:** Capacidad de Almacenamiento: La plataforma debe almacenar información histórica de hasta cinco años, con opciones de expansión de almacenamiento en caso de que sea necesario.
3. **NF-3:** Protección de Datos: Toda la información sensible debe estar encriptada.
4. **NF-4:** Control de Acceso: El sistema debe implementar un sistema de roles y permisos, de forma que cada usuario tenga acceso únicamente a las funciones correspondientes a su rol.
5. **NF-5:** Auditoría de Acciones: Cada acceso y modificación de información relevante (inscripciones, resultados de juegos, etc.) debe ser registrado en un log de auditoría para fines de control y seguridad.
6. **NF-6:** Interfaz Intuitiva: La interfaz de usuario debe ser fácil de usar y entender, permitiendo que los usuarios puedan navegar y usar el sistema sin dificultad.
7. **NF-7:** Compatibilidad Multidispositivo: El sistema debe funcionar correctamente en dispositivos de escritorio, tabletas y móviles, optimizando su diseño para cada dispositivo mediante un diseño responsivo.
8. **NF-8:** Modularidad del Código: El sistema debe tener una arquitectura modular, lo que facilita la actualización y el mantenimiento de componentes individuales sin afectar al sistema completo.
9. **NF-9:** Disponibilidad del Sistema: El sistema debe estar disponible el mayor porcentaje del tiempo.
10. **NF-10:** Copia de Seguridad: Se deben realizar copias de seguridad automáticas de los datos diariamente, y almacenarlas en un entorno seguro, con una retención de al menos 30 días para cada respaldo.
11. **NF-11:** Cumplimiento con Normativas de Privacidad: El sistema debe cumplir con las normativas de protección de datos personales.
12. **NF-12:** Integridad de la Información: La información ingresada y almacenada en el sistema no debe ser alterada sin registro, asegurando que todos los datos relevantes sean precisos y estén respaldados.

11. Otros requerimientos

- El sistema debe manejar formatos de fecha como dd/mm/yyyy para el registro en la base de datos.
- El sistema debe mantener un registro de todas las acciones críticas realizadas por los usuarios, tales como inscripciones, actualizaciones de resultados, cambios en permisos de acceso, entre otras, para auditoría y resolución de conflictos.

12. Glosario

- API (Application Programming Interface): Interfaz de programación de aplicaciones que permite la comunicación entre diferentes aplicaciones de software.
- Autenticación: Proceso de verificación de la identidad de un usuario antes de permitirle el acceso al sistema.
- Base de Datos Relacional: Sistema de almacenamiento de datos que organiza la información en tablas relacionadas entre sí mediante claves.
- CI (Cédula de Identidad): Documento oficial de identificación utilizado en muchos países, requerido para verificar la identidad de los estudiantes en el sistema.
- Frontend: Parte de la aplicación que interactúa directamente con el usuario, generalmente implementada en HTML, CSS, y JavaScript.
- HTTPS (HyperText Transfer Protocol Secure): Protocolo de transferencia de hipertexto seguro que protege la comunicación en línea mediante encriptación SSL/TLS.
- IA (Inteligencia Artificial): Área de la informática que desarrolla sistemas capaces de realizar tareas que normalmente requieren inteligencia humana, como el análisis de rendimiento en esta plataforma.
- JWT (JSON Web Token): Formato estándar para la creación de tokens de autenticación que permiten gestionar sesiones de usuario de forma segura.
- Programación Lineal: Técnica de optimización utilizada para resolver problemas de asignación de recursos bajo ciertas restricciones, aplicada en la plataforma para distribuir horarios y espacios deportivos.