|  |  |
| --- | --- |
| TECNICATURA SUPERIOR EN  DISEÑO WEB Y APLICACIONES MÓVILES |  |
| Espacio curricular  Proyecto Integrador I | |
| Especificación de requisitos de software  Proyecto: MiCoachFit | |
| Estudiantes: BALBASTRO, Carlos Eduardo; BOSQUE, Rubén; Darío; HEREDIA, Eric Victor Hugo; MOYANO, Maximino | |

**Ficha del documento**

| **Fecha** | **Revisión** | **Autor** | **Referencia de Cambio** |
| --- | --- | --- | --- |
| 18/08/2025 | 1 | Todos los integrantes | Se comienza la construcción del documento |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Contenido**

[**1 Introducción (Eric) 3**](#_in6u413g4t16)

[1.1 Propósito (Eric) 4](#_ulwwaj1qaelx)

[1.2 Alcance (Dario) 4](#_gnenrk6r2ga7)

[1.3 Personal involucrado (Dario) 4](#_f1z9v5sif5s6)

[1.4 Definiciones, acrónimos y abreviaturas(Tiago) 5](#_kdvf5kzd3515)

[1.5 Referencias 6](#_fabeewe6ezbw)

[1.6 Resumen(Tiago) 6](#_1098xhn8stjh)

[**2 Descripción general 6**](#_wq6h6bedr751)

[2.1 Perspectiva del producto(Kiara) 6](#_xdtp0ho2x0ox)

[2.2 Características de los usuarios (Maximino) 7](#_qmr43txnnuxy)

[2.3 Restricciones(Maximino) 7](#_goi1n8doz35m)

[**3 Requisitos específicos 7**](#_9ay4hnj95ctt)

[3.1 Requerimientos Funcionales(Kiara) 7](#_7tzl5trdg244)

[3.2 Requisitos No Funcionales(Carlos) 8](#_373p7u57xd06)

[**Product Backlog(Carlos) 9**](#_yoc72aumphes)

[Historia de Usuario 9](#_g36etvwq7jkt)

[Criterios de Aceptación 9](#_533kkls0j6wq)

[Historia de Usuario 9](#_t0pyrduwq0e3)

[Criterios de Aceptación 10](#_nqd3i5kk7mpl)

[**Sprints 10**](#_fk3uli2cur7c)

[**Anexo I (Carlos) 13**](#_8a5eks7m3bjn)

[DER link 13](#_nmejlu1z7m3n)

# Introducción (Eric)

Este documento es una Especificación de Requisitos Software (ERS) para el Sistema de información para la gestión de la App Web MiCoachFit. Esta especificación se ha estructurado basándose en las directrices dadas por el estándar IEEE Práctica Recomendada para Especificaciones de Requisitos Software ANSI/IEEE 830, 1998, en combinación con las prácticas propias de las metodologías ágiles para garantizar que los requisitos y la evolución del proyecto.

## **Propósito (Eric)**

El presente documento tiene como propósito definir las especificaciones funcionales, para el desarrollo de una aplicación web que permitirá simplificar la gestión de clases grupales coordinadas por entrenadores personales. La app web busca brindar a los coaches una herramienta intuitiva que les permita organizar sus horarios, controlar la capacidad de sus clases y ofrecer a sus clientes una experiencia de reserva de turnos fluida y sencilla.

## Alcance (Dario)

**Esta especificación de requisitos está dirigida al usuario del sistema**, principalmente entrenadores personales que gestionan clases grupales, y tiene como propósito definir las funcionalidades esenciales de la aplicación web *MiCoachFit*. El sistema permitirá a los coaches organizar sus horarios, establecer la capacidad máxima de cada clase y facilitar a sus clientes la reserva de turnos de manera autónoma y sencilla.

El alcance de esta aplicación incluye:

* La creación y gestión de clases grupales por parte del entrenador.
* La configuración de cupos máximos por clase.
* Un sistema de reservas accesible para los alumnos.
* La visualización de horarios disponibles.
* La notificación de confirmaciones o cancelaciones de turnos.

Quedan fuera del alcance en esta etapa funcionalidades como pagos en línea, seguimiento de rendimiento físico o integración con dispositivos externos. Estas podrían considerarse en futuras versiones.

## **Personal involucrado (**Dario**)**

| **Nombre** | Carlos Balbastro |
| --- | --- |
| **Rol** | Scrum Master |
| **Categoría Profesional** | Estudiante |
| **Responsabilidad** | Desarrollo |
| **Información de contacto** | caebalbastro@gmail.com |

| **Nombre** | Dario Bosque |
| --- | --- |
| **Rol** | Developer |
| **Categoría Profesional** | Estudiante |
| **Responsabilidad** | Desarrollador |
| **Información de contacto** | dariobosque@gmail.com |

| **Nombre** | Eric Heredia |
| --- | --- |
| **Rol** | Developer |
| **Categoría Profesional** | Estudiante |
| **Responsabilidad** | Desarrollador |
| **Información de contacto** | ericvheredia@gmail.com |

| **Nombre** | Maximino Moyano |
| --- | --- |
| **Rol** | Developer |
| **Categoría Profesional** | Estudiante |
| **Responsabilidad** | Desarrollador |
| **Información de contacto** | maxgelmax21@gmail.com |

| **Nombre** | Kiara Fernandez |
| --- | --- |
| **Rol** | Developer |
| **Categoría Profesional** | Estudiante |
| **Responsabilidad** | Desarrollo |
| **Información de contacto** | kiarifer2005@gmail.com |

| **Nombre** | Tiago Nievas |
| --- | --- |
| **Rol** | Developer |
| **Categoría Profesional** | Estudiante |
| **Responsabilidad** | Desarrollo |
| **Información de contacto** | tiagonievas30@gmail.com |

## **Definiciones, acrónimos y abreviaturas(Tiago)**

| ***Nombre*** | ***Descripción*** |
| --- | --- |
| **Usuario** | Persona que usará el sistema para gestionar procesos |
| **SIS-I** | Sistema de Información Web para la Gestión de Procesos Administrativos y Académicos |
| **ERS** | Especificación de Requisitos Software |
| **RF** | Requerimiento Funcional |
| **RNF** | Requerimiento No Funcional |
| **FTP** | Protocolo de Transferencia de Archivos |
| **Moodle** | Aula Virtual |
| **MCF** | App Web MiCoachFit |
| **DER** | Diagrama Entidad de Relación |
| **ARIA** | Accessible Rich Internet Applications |
| **NVDA** | NonVisual Desktop Access |
| **JAWS** | Job Access With Speech |

## **Referencias**

| **Título del Documento** | **Referencia** |
| --- | --- |
| Standard IEEE 830 - 1998 | IEEE |

## **Resumen(Tiago)**

Este documento orientado con el estándar IEEE 830 busca detallar las funciones de la app MiCoachFit (MCF) que permite a personal trainers gestionar de mejor manera clases grupales brindándoles herramientas para visualizar horarios disponibles, número de participantes y gestión de turnos usuarios.

# Descripción general

## **Perspectiva del producto(Kiara)**

MiCoachFit es una aplicación web innovadora diseñada para simplificar la gestión de clases grupales en gimnasios y entrenadores personales. El producto brinda a los coaches una herramienta intuitiva que les permite organizar sus horarios, controlar la capacidad de sus clases y ofrecer a sus clientes una experiencia de reserva de turnos rápida, fluida y sencilla

## **Características de los usuarios (Maximino)**

| **Tipo de usuario** | Usuarios Primarios |
| --- | --- |
| **Formación** | Entrenadores personales |
| **Actividades** | Ofrecen clases grupales y buscan limitar los participantes para mejor seguimiento. |

| **Tipo de usuario** | Usuario Secundario |
| --- | --- |
| **Formación** | Personas |
| **Actividades** | Asistencia a las clases y reservan sus turnos a través de la plataforma/app. |

*.*

## **Restricciones(Maximino)**

* Interfaz para ser usada con internet.
* Lenguajes y tecnologías en uso: HTML,CSS,JavaScript,BootStrap y Python.

# Requisitos específicos

## **Requerimientos Funcionales(Kiara)**

Las funcionalidades se centrarán principalmente en las necesidades del entrenador:

Gestión de Horarios de Clases

* + Creación de clases grupales con detalles como tipo de entrenamiento, duración, lugar y capacidad máxima de los alumnos.
  + Definición de días y horarios de las clases.
  + Posibilidad de editar o cancelar clases.

Gestión de Reservas

* + Visualización clara de los alumnos que hicieron su reserva en cada clase.
  + Lista de espera automática cuando una clase alcanza su capacidad máxima. Si un alumno cancela su reserva, toma el lugar quien se encuentre en la lista de espera. (opcional)

Gestión de Alumnos

* + Registro de alumnos con información básica (nombre y apellido, correo, teléfono)
  + Historial de reservas por alumno.
  + Posibilidad de comunicarse con alumnos (mensajes integrando con otros medios como whatsapp)
  + Mensajes de notificación o alerta para dar aviso de eventualidades (clases suspendidas, promociones, otros)

## Requisitos No Funcionales(Carlos)

**Diseño Responsivo**

* La aplicación debe adaptarse automáticamente a distintos tamaños de pantalla (PC, tablet, smartphone).
* Se utilizarán tecnologías como CSS y frameworks como Bootstrap para garantizar una experiencia fluida en cualquier dispositivo.

**Rendimiento**

* El sistema debe cargar las páginas en menos de 3 segundos bajo condiciones normales de red.
* Las operaciones de reserva, edición de clases y autenticación deben ejecutarse en tiempo real sin demoras perceptibles.

**Seguridad**

* Toda la comunicación entre cliente y servidor debe estar cifrada mediante HTTPS.
* Los datos personales y credenciales de los usuarios serán almacenados de forma segura utilizando algoritmos de hash y encriptación.
* Se aplicarán políticas de control de acceso para evitar que usuarios no autorizados accedan a funciones restringidas.

**Accesibilidad para Personas con Discapacidad Visual**

* La interfaz será compatible con lectores de pantalla como NVDA y JAWS.
* Se utilizarán etiquetas ARIA y estructuras semánticas para facilitar la navegación.
* Se garantizará un contraste adecuado entre texto y fondo, y se ofrecerá la posibilidad de aumentar el tamaño de fuente.

**Modularidad**

* La aplicación debe estar diseñada de forma modular, permitiendo que sus componentes (gestión de clases, reservas, autenticación, etc.) estén desacoplados y sean independientes.
* Esto facilitará el mantenimiento, la escalabilidad y la incorporación de nuevas funcionalidades sin afectar al sistema completo.
* Cada módulo debe tener una responsabilidad clara y comunicarse con otros mediante interfaces bien definidas.

# Product Backlog(Carlos)

* Hacer Wireframe
* Trasladar el wireframe a mano alzada a Figma
* Continuar pruebas de Consola(Escritorio)
* Repasar/Mejorar si es necesario las DER
* Realizar Maquetas en HTML, CSS, BOOTSTRAP, JS

### Historia de Usuario

Como **profesor de gimnasia**, quiero **tener un calendario digital donde pueda ver todas mis clases grupales programadas**, para poder **organizar mi semana de forma eficiente y no tener conflictos de horario**.

### Criterios de Aceptación

Para que esta historia de usuario sea considerada completa, la aplicación debe permitir:

1. El profesor puede **ver todas sus clases programadas** en un formato de calendario semanal o mensual.
2. Cada clase en el calendario muestra información clave: **nombre de la clase**, **horario**, **ubicación** (ej. "Gimnasio", "Parque") y **número de cupos disponibles**.
3. El profesor puede **agregar una nueva clase** directamente desde el calendario, especificando los detalles necesarios.
4. El profesor puede **editar o cancelar una clase existente** con un solo clic, y el sistema actualiza automáticamente la disponibilidad para los clientes.
5. El calendario tiene una **vista clara y fácil de leer** en dispositivos móviles y de escritorio.

### Historia de Usuario

Como **cliente de un entrenador personal**, quiero **poder ver su horario de clases disponibles y reservar mi lugar en una clase específica**, para **asegurarse un cupo sin tener que contactar directamente**.

### Criterios de Aceptación

Para que esta historia de usuario se considera completada, la aplicación debe permitir:

1. El cliente puede **navegar por un calendario** que muestra las próximas clases grupales.
2. Cada clase en el calendario muestra el **nombre de la clase**, el **horario**, la **ubicación** y el **número de cupos restantes**.
3. El cliente puede **seleccionar una clase y reservar un cupo** con un proceso simple (ej. un solo clic).
4. La aplicación **confirma la reserva** y envía al cliente una notificación (ej. un correo electrónico o una notificación en la app) con los detalles de la clase.
5. El cliente tiene una **vista de "Mis Reservas"** donde puede ver todas sus clases futuras y, si es necesario, cancelar su cupo.

# Sprints

| **N° de sprint** | 00 |
| --- | --- |
| **Sprint Backlog** | US01  US02  TK01  TK02  -  -  - |
| **Responsabilidades** | **Seleccionar los ítems** del Product Backlog para el sprint.  **Dividir los ítems** del Product Backlog en tareas más pequeñas y gestionables.  **Completar las tareas** que se han comprometido a realizar.  **Colaborar** entre ellos para asegurarse de que el trabajo se haga de manera eficiente.  **Actualizar el estado** de las tareas en el Sprint Backlog.  **Mantener la calidad** del trabajo que están realizando. |
| **Calendario** | Fecha Límite para el Objetivo del Sprint |
| **Inconvenientes:** Algunos no puede asistir a las reuniones diarias del sprint porque tiene otra reunión fija a la misma hora | |

| **N° de sprint** | 01 |
| --- | --- |
| **Sprint Backlog** | US10  US11  US12  -  -  - |
| **Responsabilidades** |  |
| **Calendario** |  |
| **Inconvenientes:** | |

| **N° de sprint** | 02 |
| --- | --- |
| **Sprint Backlog** | US20  US21  US22  -  -  - |
| **Responsabilidades** |  |

# Anexo I (Carlos)

## DER link

[diagramas-micoachfit.drawio (2).pdf](https://drive.google.com/file/d/1zXuVUrIn9Id7bTHzY_3oDzCLmqE8BlD5/view?usp=sharing)

## 

**Diagrama de Clase**

[Documentación-Programación-MiCoachFit.pdf](https://drive.google.com/file/d/14QO-_GQ1Z-trLJoiIMmzVgl4dzfYi0sV/view?usp=sharing)