



### Entrega 03

#### 1. Clasificación y estimación.

Usamos las siguientes convenciones:

- Requisito funcional 1 (RF\_ 1)
- Requisito NO funcional 1 (RNF\_ 1)
- Administración de socios y membresías (ADM)
- Control de acceso y asistencia (ACC)
- Pagos y alertas (PAY)
- Catálogos y gestión de datos (CAT)
- Planificación de rutinas (RUT)
- Planificación nutricional (NUT)
- Progreso, analítica y recomendaciones (PRG)
- Seguridad, respaldo y auditoría (SEC)

Requisito			Estimación	Argumento
	RF_1	<b>ADM-R1.1:</b> El Sistema debe permitir a la Recepción/Administrador registrar nuevos socios capturando todos los campos obligatorios: Identificación, Nombre Completo, Teléfono, Correo Electrónico, Fecha de Nacimiento y Consentimiento de datos.	2	Es un formulario de registro estándar (CRUD), pero la validación de campos obligatorios, la interfaz inicial (mockup) y el modelado de datos consumen tiempo. Tarea rutinaria.
		<b>ADM-R1.2:</b> El Sistema permite a la Recepción/Administrador incluir y visualizar en el perfil		Extensión menor del formulario principal. Implica solo añadir dos campos



<b>MUST</b>	<b>RF_2</b>	del socio información opcional, como el estado de salud y una nota de seguimiento.	1	opcionales al modelo y a la interfaz de usuario. Es una tarea de verificación sencilla.
	<b>RF_3</b>	<b>ADM-R1.3:</b> El Sistema debe facilitar a la Recepción/Administrador la edición y actualización de los datos ya registrados de cualquier socio para mantener la información vigente.	2	Requiere implementar una vista de detalle/edición y la lógica para actualizar la BD sin generar inconsistencias. Es la parte final del CRUD de socios.
	<b>RF_4</b>	<b>ADM-R2.1:</b> El Sistema DEBE permitir al Administrador crear, modificar y administrar el catálogo de Tipos de Membresías, definiendo nombre, precio, duración (días/meses) y beneficios asociados.	2	Implica un CRUD simple para la tabla de Membresías. Es una tarea de rutina que requiere poco esfuerzo, pero es crítica para la estructura de la BD.
	<b>RF_5</b>	<b>ADM-R2.2:</b> El Sistema debe realizar el cálculo automático de la Fecha de Finalización al momento de que la Recepción asigne una membresía a un socio, sumando la duración	3	Requiere la lógica para validar la existencia del socio y calcular automáticamente la fecha_finalización con precisión. El cálculo automático añade complejidad media.



		establecida al día de inicio.		
	<b>RF_6</b>	<b>ACC-R3.1:</b> El Sistema debe registrar el evento de asistencia (entrada/salida) de un socio al interactuar con el terminal (ej. lector de código de barras/QR), vinculando el registro al tipo de acceso y la hora exacta.	3	Implica diseñar una interfaz de búsqueda rápida y la lógica para registrar el evento de asistencia. El manejo de la "terminal" (aunque sea simulada) añade una capa de interacción.
	<b>RF_7</b>	<b>ACC-R3.2:</b> El Sistema debe realizar la validación del estado de la membresía del socio en tiempo real, e impedir el acceso si no está Activa, notificando a la Recepción el motivo exacto (ej. Morosa o Expirada).	5	Tarea Crítica y Compleja. Requiere lógica de reglas de negocio avanzada para consultar el estado (Activa, Morosa, Expirada) y mostrar el motivo. Esto debe ser súper rápido (requisito no funcional de rendimiento).
	<b>RF_8</b>	<b>PAY-R6.1:</b> El Sistema debe ejecutar la actualización automática del estado de una membresía a "Activo" inmediatamente después de que la Recepción registre un pago por el	3	Tarea de complejidad media. Implica asegurar una transacción exitosa (registro de pago + actualización de estado) que debe ser atómica, lo cual es crítico para las reglas de negocio.



		monto total pendiente.		
	RF_9	<b>PAY-R6.2:</b> El Sistema debe registrar la deuda restante y mantener el estado de la membresía como Morosa cuando la Recepción registra un pago parcial inferior al monto total pendiente.	4	La lógica de pagos parciales es compleja. Requiere calcular el nuevo saldo pendiente, mantener el historial de la deuda, y actualizar el estado de la membresía a Morosa (membresía con pago pendiente que impide el acceso). Esto demanda un esfuerzo considerable en el diseño del modelo de la Base de Datos para asegurar la integridad transaccional.
	RF_10	<b>PAY-R6.3:</b> El Sistema debe validar que el monto de pago ingresado por la Recepción no exceda el valor total de la membresía ni sea negativo, garantizando la integridad de la transacción.	1	Validación simple (Monto $\geq 0$ y Monto $\leq$ Pendiente). Es una tarea de rutina en la capa de interfaz de usuario/servicios.
	RNF_1	<b>SEC-R16.1:</b> El Sistema debe exigir la autenticación de los Usuarios Internos (Administrador, Entrenador, Recepción) mediante credenciales de usuario y contraseña para	4	Complejo por el conocimiento del equipo. Requiere implementar hashing seguro de contraseñas, el flujo de inicio de sesión y la gestión de la sesión, lo cual es difícil porque el equipo no sabe hacer un auth con token bearer o el lenguaje (como en el ejemplo).



		acceder a cualquier funcionalidad.		
	RF_11	<b>SEC-R16.2:</b> El Sistema debe gestionar la asignación de roles (Administrador, Entrenador, Recepción) a los Usuarios Internos y aplicar el control de permisos asociados a cada rol, restringiendo el acceso a módulos no autorizados.	4	Implica diseñar un sistema de autorización robusto (RBAC) que impacta casi todas las pantallas, lo que requiere un esfuerzo considerable en el framework que se elija.
	RNF_2	<b>SEC-R16.3:</b> El Sistema debe ejecutar la expiración automática de la sesión de los Usuarios Internos tras un período de inactividad definido, por motivos de seguridad.	3	Implica implementar la lógica de timeout y el manejo de tokens/sesiones, lo cual es un esfuerzo medio que requiere cuidado en la capa de seguridad.
	RF_12	<b>PAY-R5:</b> el sistema debería generar un recibo de pago (impreso o digital) para el socio cada vez que la recepción complete una transacción.	2	Es una tarea de reporting rutinaria y sencilla. Requiere poco esfuerzo de maquetación y una función de exportación (ej. a PDF), lo cual es de complejidad baja a media.



SHOULD	RF_13	<b>ACC-R4:</b> el sistema debería facilitar al administrador y al entrenador la visualización de reportes y gráficas de la asistencia diaria, semanal o mensual.	3	La complejidad reside en la consulta a la BD por rangos de tiempo y en la necesidad de usar una librería de gráficos (charts), lo cual consume más tiempo que un CRUD simple.
	RF_14	<b>RUT-R9:</b> el sistema debería permitir al entrenador construir y planificar las rutinas de entrenamiento de un socio usando una interfaz de tipo <i>builder</i> (arrastrar-y-soltar).	4	Complejidad alta en la interfaz. Diseñar una UI de arrastrar-y-soltar (builder) es un esfuerzo significativo que implica programación avanzada del front-end para ser funcional y usable.
	RF_15	<b>PRG-R14:</b> el sistema debería permitir al entrenador y al socio el registro histórico de progreso (peso, medidas corporales) y mostrar su evolución a través de gráficas exportables.	4	La dificultad es similar a la de ACC-R4, pero requiere un modelo de datos histórico más robusto (para peso y medidas) y librerías de visualización de datos avanzadas para mostrar la evolución al socio.



	RNF_3	<b>SEC-R17:</b> el sistema debería ofrecer al administrador la funcionalidad de configuración y ejecución de respaldos locales, incluyendo la gestión de intervalos, retención y restauración de datos.	5	Tarea compleja. Requiere conocimiento de cómo interactuar con el sistema de archivos local para la gestión de respaldos (intervalos, retención) y la restauración de la base de datos local, lo cual es crítico para la confiabilidad.
COULD	RF_16	<b>CAT-R8:</b> el sistema podría permitir al administrador y al entrenador gestionar un catálogo local de alimentos, manteniendo el versionamiento y la auditoría de los cambios realizados..	4	El CRUD es bajo (1). Pero la funcionalidad de versionamiento y auditoría de los cambios es compleja, ya que requiere añadir timestamps y lógica de registro de actividad a la tabla, aumentando el esfuerzo en el diseño de la BD.
	RF_17	<b>RUT-R10:</b> el sistema podría aplicar la validación de límites de series, repeticiones y peso recomendados en la rutina, alertando al entrenador si se exceden las reglas de negocio.	2	Es una capa de validación de negocio. Es un esfuerzo bajo que se puede implementar con validadores simples en la capa de servicio/interfaz (e.g., if (series > 8) then alert).



	RF_18	<b>RUT-R11:</b> el sistema podría permitir al entrenador guardar rutinas creadas como plantillas, y registrar la ejecución de la sesión del socio, capturando los resultados obtenidos por ejercicio.	3	La gestión de plantillas es un CRUD sencillo, pero el registro de la ejecución de la sesión requiere un nuevo formulario y modelar la relación entre Rutina y Ejecución, lo que añade complejidad media.
	RF_19	<b>NUT-R12:</b> el sistema podría asistir al entrenador en la creación del plan nutricional de un socio, calculando y mostrando automáticamente los valores totales de energía y macronutrientes por día y semana.	3	El esfuerzo es medio. El requisito implica implementar un cálculo matemático (suma de macronutrientes y calorías del catálogo) en tiempo real para el entrenador. Se requiere investigación sobre las fórmulas.
WON'T	RF_20	<b>NUT-R13:</b> El sistema permitirá sustituir un alimento en una comida por otro del catálogo siempre que la sustitución mantenga el objetivo calórico diario dentro de $\pm 5\%$ . Si la sustitución rompe el margen, mostrar alerta y proponer ajuste automático de porciones para	5	Tarea extremadamente compleja. El desarrollo de un algoritmo de optimización que sustituya alimentos, recalcule y ajuste porciones dentro de un margen $\pm 5\%$ es un esfuerzo de 5+. Se desprioriza por el alto riesgo técnico.





		alcanzar el objetivo dentro del $\pm 5\%$ .		
	RF_21	<b>PRG-R15:</b> El sistema identifica estancamientos en el progreso de levantamientos clave y genera recomendaciones automáticas para el entrenador, quien registra la acción tomada.	5	Tarea extremadamente compleja. Requiere la implementación de lógica predictiva o analítica para definir qué es un "estancamiento" y generar recomendaciones automáticas. Es un esfuerzo de 5+. Se desprioriza por la complejidad y falta de tiempo.