



Instituto Tecnológico Colonia CTC

Analista Programador

Programación III

**1.er Obligatorio**

**Sistema gestor de gimnasios.**

Docente: Ignacio Oneto

Alumna: Melanie Machín

**2024**

## Declaración de autoría

Por la presente, yo, Melanie Machín, estudiante del instituto Tecnológico de Colonia CTC, certifico que:

- El presente proyecto fue elaborado durante mi participación en la asignatura “Programación III”, bajo la supervisión del docente Ignacio Oneto.
- Las ideas y desarrollos presentados en este trabajo se basan en los conceptos aprendidos en clase y en los materiales proporcionados por el docente, así como en conocimientos y recursos obtenidos más allá del ámbito académico institucional.
- Declaro que esta obra es original y de mi exclusiva autoría, y no ha sido sometida previamente a ninguna otra evaluación.
- Reconozco que la propiedad intelectual de este trabajo pertenece al instituto Tecnológico de Colonia.

---

Melanie Machín.

# Contenido

|   |   |
|---|---|
| LECTURA DEL PROBLEMA.....                 | 4 |
| REQUERIMIENTOS .....                      | 6 |
| <i>Actores.....</i>                       | 6 |
| <i>Funcionalidades del sistema.....</i>   | 6 |
| <i>Requerimientos no funcionales.....</i> | 6 |
| MODELO DE DOMINIO .....                   | 7 |
| DIAGRAMA CASO DE USO .....                | 8 |
| CASO DE USO NARRATIVO .....               | 9 |

## Lectura del problema

La red de gimnasios "24" ha reconocido la necesidad de modernizar y optimizar la gestión de su negocio. Para lograr este objetivo, están buscando implementar un sistema integral que les permita administrar eficientemente todos los aspectos relacionados con el negocio, incluyendo la gestión de socios, locales y máquinas.

Los socios, que representan los usuarios registrados en los gimnasios, desempeñan un papel fundamental en este sistema. Se requiere almacenar información detallada sobre cada socio, incluyendo su nombre, tipo de usuario (que puede ser 'básico', 'estándar' o 'premium'), número de teléfono, correo electrónico y el local al que están afiliados.

Los locales son los distintos lugares físicos a los que los socios asisten para realizar sus actividades de entrenamiento. Para cada local, se debe mantener un registro que incluya el nombre del local, la ciudad, la dirección física, el número de teléfono y los detalles del responsable del local, que incluyen su nombre y número de teléfono.

Las máquinas de ejercicio son un componente clave en cada local y tienen una relación directa con ellos. Se debe registrar información detallada sobre cada máquina, incluyendo el local al que pertenece, la fecha de compra, el precio de compra, la vida útil en años, el tipo de máquina y su disponibilidad actual (es decir, si está disponible para uso).

Es importante destacar que el sistema debe proporcionar funcionalidades completas de administración (CRUD) para todas estas entidades: socios, locales y máquinas. Además, se requieren consultas específicas para facilitar la gestión del negocio, tales como:

- a. Permitir a los administradores filtrar y visualizar a los socios según su tipo de usuario (estándar o premium).
- b. Identificar los tipos de máquinas disponibles en todos los locales, así como la cantidad de máquinas de cada tipo, con la opción de filtrar por local.
- c. Calcular automáticamente los años de vida útil restantes para una máquina específica.
- d. Permitir la ordenación de las máquinas por fecha de compra, tanto ascendente como descendente.

Para abordar este problema en el desarrollo de software utilizando el lenguaje de programación C#, se debe llevar a cabo un análisis exhaustivo de los requerimientos

funcionales y no funcionales correspondientes. Además, se debe diseñar una estructura de clases coherente y eficiente, aplicando las mejores prácticas y métodos de desarrollo de software.

Para visualizar y comprender mejor la estructura del sistema, se debe crear un diagrama de clases que represente los modelos de datos involucrados, incluyendo clases para socios, locales y máquinas, así como cualquier otra clase necesaria para el funcionamiento del sistema.

Finalmente, se debe desarrollar el código en C# que permita implementar la solución propuesta, utilizando una base de datos relacional proporcionada en BDII para almacenar y gestionar la información de manera eficiente. Este código debe ser modular, fácil de mantener y escalable para futuras expansiones del sistema.

## Requerimientos

### Actores

- Responsable:
- Administrador:

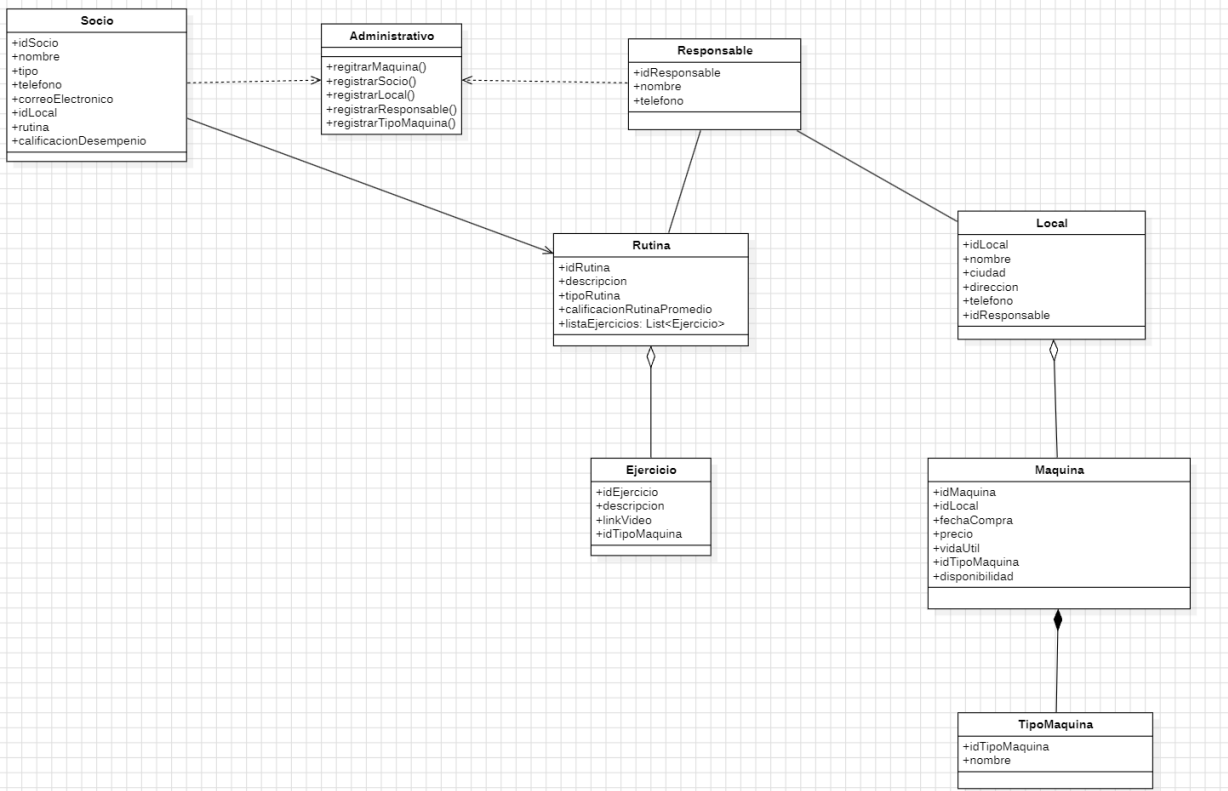
### Funcionalidades del sistema

- Socio:
  - Registrar Socio
  - Consultar socio
  - Actualizar socio
  - Eliminar socio
- Local:
  - Registrar local
  - Consultar local
  - Actualizar local
  - Eliminar local
- Responsable:
  - Registrar responsable
  - Consultar responsable
  - Actualizar responsable
  - Eliminar responsable
- Maquina:
  - Registrar maquina
  - Consultar maquina
  - Actualizar maquina
  - Eliminar maquina
- Tipo maquina
  - Registrar tipo maquina
  - Consultar tipo maquina
  - Actualizar tipo maquina
  - Eliminar tipo maquina

### Requerimientos no funcionales

- El sistema debe estar desarrollado en lenguaje de programación C#.
- La persistencia de los datos será mediante el sistema de gestión de base de datos.
- SQL Server.

# Modelo de dominio



## Diagrama caso de uso





Caso de uso narrativo

|  |                |
|--|----------------|
| Identificador  |                |
| Caso de uso  | eliminar socio |
| Fecha  | 15/05/2024     |
| Descripción: permite eliminar un socio                   |                |
| Actor/es: administrativo                                 |                |
| Precondiciones: GestionandoSocio                         |                |
| Actor/es:  |                |
| Flujo normal: Borrar socio.                              |                |
| 1- Sistema permite ingresar idSocio.                     |                |
| 2- Sistema permite solicitar borrar socio.               |                |
| 3- administrador ingresa idSocio.                        |                |
| 4- dministrador solicita borrar socio.                   |                |
|  |                |
| Flujo alternativo/s: Socio no existe.                    |                |
| 1- Sistema muestra no existe socio con el id registrado. |                |
| 2- ir a flujo normal paso 2.                             |                |
|  |                |
|  |                |
|  |                |
| Flujo Excepcional/es:                                    |                |
|  |                |
|  |                |
|  |                |
|  |                |
| Pos Condiciones:   |                |
| GestionandoSocio   |                |
|  |                |

|  |                |
|--|----------------|
| Identificador  |                |
| Caso de uso  | eliminar local |
| Fecha  | 15/05/2024     |
| Descripción: permite eliminar un local                   |                |
| Actor/es: administrativo                                 |                |
| Precondiciones: GestionarLocal                           |                |
| Actor/es:  |                |
| Flujo normal: Borrar Local.                              |                |
| 1- Sistema permite ingresar idLocal.                     |                |
| 2- Sistema permite solicitar borrar local.               |                |
| 3- administrador ingresa idLocal.                        |                |
| 4- Administrador solicita borrar Local.                  |                |
|  |                |
| Flujo alternativo/s: Local no existe.                    |                |
| 1- Sistema muestra no existe local con el id registrado. |                |
| 2- ir a flujo normal paso 2.                             |                |
|  |                |
|  |                |
|  |                |
| Flujo Excepcional/es:                                    |                |
|  |                |
|  |                |
|  |                |
|  |                |
| Pos Condiciones:   |                |
| GestionarLocal   |                |