

Instituto Tecnológico Colonia CTC

Analista Programador
Programacion 3

Obligatorio Programación 3

Docente: Ignacio Onetto
Alumno: Sebastián Álvarez

Indice:

Letra del problema	3
Requerimientos Funcionales	6
Diagrama de Clases	7
Diagrama de casos de uso	8
Casos de uso narrativo	9

Letra del problema

Obligatorio - Parte 1 | Programación 3

Fecha de inicio: 15/04/2023.

Fecha de entrega: A definir.

Descripción del Problema:

La red de gimnasios "24" necesita modernizarse y gestionar su negocio de manera eficiente. Para ello, requiere un sistema que permita gestionar el manejo del negocio, incluyendo socios, locales y máquinas.

Los **socios** son aquellos usuarios que se registran y administran desde el momento en el que una persona se afilia. Se desea contar con nombre, tipo, teléfono, mail y el local al que se inscribe (el tipo de usuario puede ser *'estándar'* o *'premium'*).

Los **locales** son los distintos lugares físicos a los que los socios acuden. De los mismos es requerido administrar nombre, ciudad, dirección, teléfono y un responsable (el responsable es una persona de la cual se registra nombre y teléfono).

Las **máquinas** se encuentran en cada local y tienen una relación directa con los mismos. De ellas se desea almacenar el local al que pertenece, fecha en la que fue comprada, precio por el que fue comprada, su vida útil, el tipo de máquina y si se encuentra disponible.

- La vida útil se contabiliza en años.
- El tipo de máquina debe ser a su vez administrable (CRUD), (por ejemplo: cinta de correr, bicicleta estática, prensa de piernas, etc).

Se pide:

1. Para todas estas entidades es **mandatorio** proporcionar funcionalidades de administración (CRUD).

2. Consultas:

- a. El sistema debe permitir a los administradores filtrar y visualizar a los socios según su tipo de usuario (estándar o premium).
- b. Los administradores necesitan identificar qué tipos de máquinas existen en todos los locales y cuántas de cada tipo hay. Se debe permitir a su vez la posibilidad de filtrar por local.
- c. El sistema debe permitir calcular los años de vida útil restantes de una máquina en específico.
- d. Debe ser posible ordenar por fecha de compra tanto ascendente como descendentemente.

Consultas adicionales (para 3 integrantes):

- El sistema debe permitir a los administradores filtrar y visualizar a los socios según el local al que pertenecen.
- El sistema debe permitir a los administradores filtrar y visualizar las máquinas según su tipo y disponibilidad.

Pasos a Seguir:**Análisis y Solución en C#:**

- Analizar el problema de gestión del gimnasio y documentar los requerimientos funcionales y no funcionales correspondientes.
- Diseñar una estructura de clases acorde a la situación y aplicando buenas prácticas y métodos.

Diagrama de Clases:

Crear un diagrama de clases que represente los modelos incluidos en el sistema.

Código:

Desarrollar el código en C# que, una vez ejecutado, permita implementar la solución al problema planteado utilizando la base de datos proporcionada.

Entrega - Parte 1 (15/05/2024):**1. Documentación en formato PDF:**

- a. Portada del trabajo con los nombres de los integrantes del equipo.
- b. Letra del problema.
- c. Análisis y solución (relevamiento de requerimientos y no funcionales).
- d. Diagrama de clases.
- e. Diagrama general de casos de uso.
- f. Desarrollo de dos casos de uso de forma narrativa.

2. Código mediante un repositorio (GitHub):

Se deberá entregar solamente los objetos diagramados en código (Modelos).

Importante:

Utilizar la base de datos requerida por el **obligatorio de Bases de Datos 2** implementando las modificaciones que se consideren necesarias.

Requerimientos Funcionales

RF 1) El sistema deberá permitir a los administradores el registro, modificación y eliminación de socios con los siguientes campos: nombre, tipo (estándar o premium), teléfono, correo electrónico y local al que se inscribe.

RF 2) El sistema deberá permitir a los administradores el registro, modificación y eliminación de locales con los siguientes campos: nombre, ciudad, dirección, teléfono, nombre y teléfono del responsable.

RF 3) El sistema deberá permitir a los administradores el registro, modificación y eliminación de máquinas con los siguientes campos: local al que pertenece, fecha de compra, precio de compra, tipo de máquina y disponibilidad.

RF 4) El sistema deberá permitir a los administradores filtrar y visualizar socios por tipo de usuario (estándar o premium).

RF 5) El sistema deberá permitir a los administradores identificar y listar tipos de máquinas existentes en todos los locales.

RF 6) El sistema deberá permitir a los administradores la capacidad de filtrar por local. Calcular y mostrar los años de vida útil restantes de una máquina específica.

RF 7) El sistema deberá permitir a los administradores ordenar máquinas por fecha de compra, tanto en orden ascendente como descendente.

Requerimientos No Funcionales:

Las consultas y operaciones deben ejecutarse en un tiempo razonable.

Requerimientos Externos:

El sistema debe usar la Base de datos existente en un servidor conectado a una red, donde están registrados los datos del negocio.

Diagrama de Clases

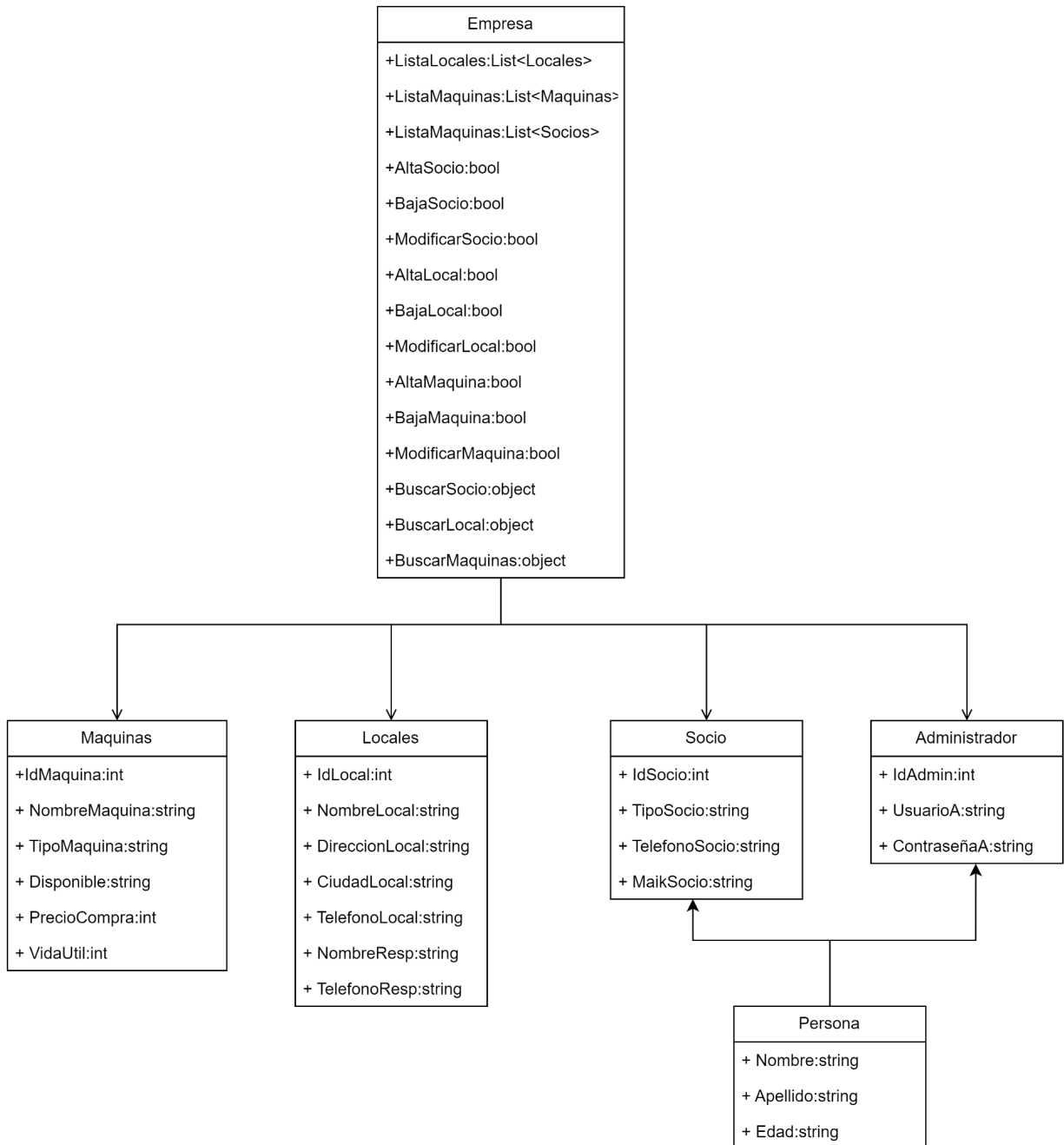
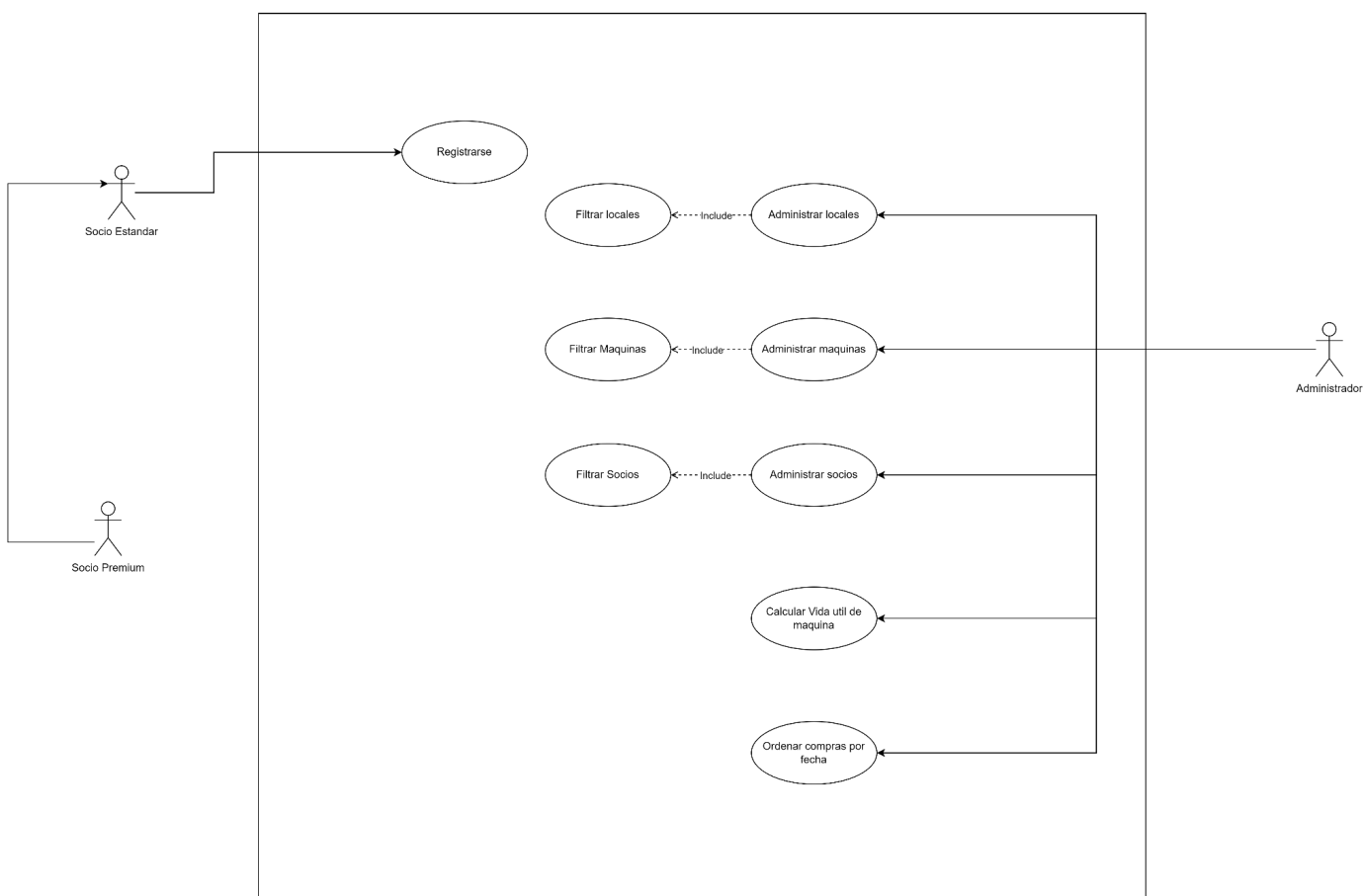


Diagrama de casos de uso



Casos de uso narrativo

Nombre	Registrarse en el gimnasio
Autor	Sebastián Álvarez
Fecha	1/05/2024
Descripción: Registrarse al gimnasio	
Precondiciones: No estar registrado	
Flujo Normal: <ol style="list-style-type: none"> 1. El actor pulsa el botón registrarse 2. El actor llena los campos pertinentes 3. El actor elige el tipo de socio que quiere ser 4. El sistema guarda los datos 	
Flujo Alternativo: <ol style="list-style-type: none"> 1. El usuario ya está registrado, por lo cual no se vuelve a guardar 	
Poscondiciones: <ol style="list-style-type: none"> 1. El usuario quedo registrado 	

Nombre	Filtrar socios
Autor	Sebastián Álvarez
Fecha	1/05/2024
Descripción: Filtrar socios registrados	
Precondiciones: Que existan socios registrados	
Flujo Normal: <ol style="list-style-type: none"> 1. El actor pulsa el botón filtrar socios 2. El actor selecciona el filtro deseado 3. El sistema toma los datos 	
Flujo Alternativo: <ol style="list-style-type: none"> 1. No existen socios 	
Poscondiciones: <ol style="list-style-type: none"> 1. El sistema muestra la lista de socios 	

Nombre	Filtrar máquinas
Autor	Sebastián Álvarez
Fecha	1/05/2024
Descripción: Filtrar máquinas compradas	
Precondiciones: Que el gimnasio tenga máquinas	
Flujo Normal: <ol style="list-style-type: none"> 1. El actor pulsa el botón filtrar máquinas 2. El actor selecciona el filtro deseado 3. El sistema toma los datos 	
Flujo Alternativo: <ol style="list-style-type: none"> 1. No existen máquinas 	
Poscondiciones: <ol style="list-style-type: none"> 1. El sistema muestra la lista de máquinas 	

Nombre	Calcular vida útil
Autor	Sebastián Álvarez
Fecha	1/05/2024
Descripción: Calcular vida útil de las máquinas	
Precondiciones: Que el gimnasio tenga maquinas	
Flujo Normal: <ol style="list-style-type: none"> 1. El actor pulsa el botón calcular vida útil de las máquinas 2. El sistema toma los datos guardados y realiza el cálculo 	
Flujo Alternativo: <ol style="list-style-type: none"> 1. No existen máquinas 	
Poscondiciones: <ol style="list-style-type: none"> 1. El sistema muestra la vida útil de las máquinas 	

Nombre	Ordenar compras
Autor	Sebastián Álvarez
Fecha	1/05/2024
Descripción: Ordena la compra de las maquinas por fecha	
Precondiciones: Que el gimnasio tenga máquinas	
Flujo Normal: <ol style="list-style-type: none">1. El actor pulsa el botón ordena la compra de las máquinas por fecha2. El sistema toma los datos guardados y ordena las máquinas	
Flujo Alternativo: <ol style="list-style-type: none">1. No existen máquinas	
Poscondiciones: <ol style="list-style-type: none">1. El sistema muestra la lista de compras por fechas	