
LICENCIATURA EN CIENCIAS DE LA COMPUTACIÓN

Ingeniería de Software II

Trabajo integrador

Grupo 4: Emiliano Germani, Paula Martinez, Martina Nahman y Enzo Palau.

Gimnasio

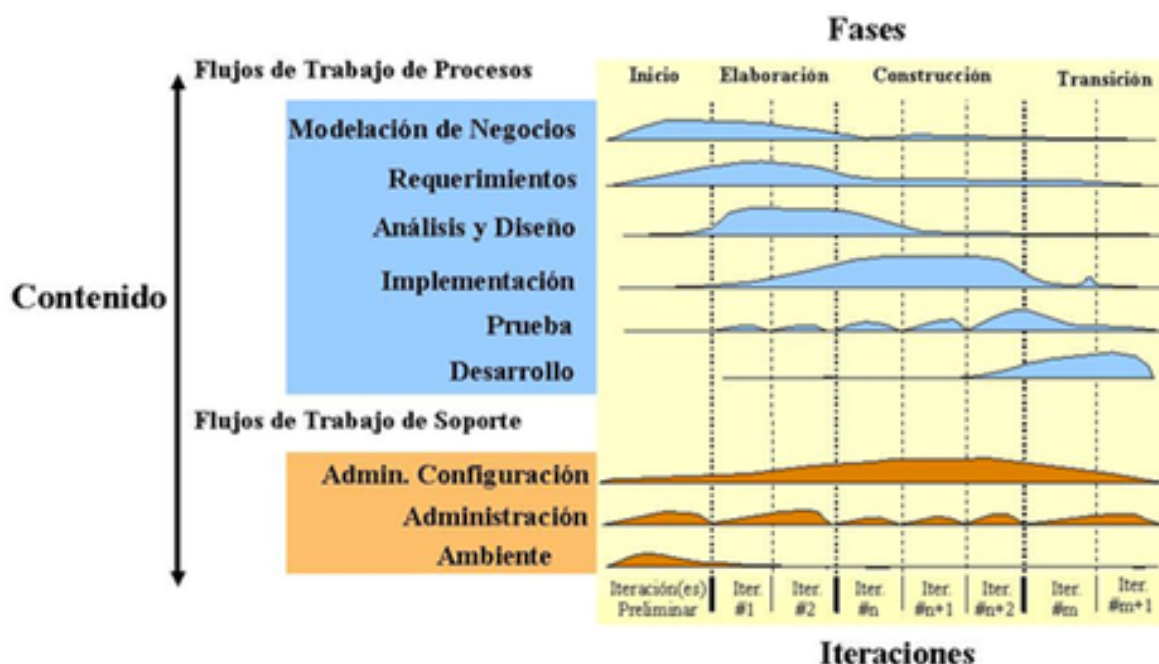
1 Introducción

En el presente informe se presenta el trabajo realizado durante dos semanas para el primer integrador de la materia "Ingeniería del software 2" 2025, en el cual se le asigna la tarea a los diferentes grupos de codificar y armar los correspondientes diagramas y documentos pertinentes para el desarrollo de un **software de gimnasio**.

2 Metodología de desarrollo

2.1 Desarrollo

Para el desarrollo de el presente trabajo se utilizó el marco de trabajo RUP (*Proceso Unificado racional*), un marco de trabajo iterativo e incremental para el desarrollo de software, no un software específico en sí mismo. Creado por Rational Software (actualmente una división de IBM), RUP es un proceso que divide un proyecto en fases y proporciona un enfoque estructurado para crear software de alta calidad que satisfaga las necesidades del usuario asegurándose de crear la documentación necesaria.



Fases del ciclo de vida: RUP define cuatro fases principales:

- Inicio: Para definir el alcance y viabilidad del proyecto.
- Elaboración: Para establecer la arquitectura del sistema y los requisitos clave.
- Construcción: Para desarrollar la mayor parte del software.
- Transición: Para desplegar el producto final y pasar a producción

2.2 Arquitecturas, patrones y tecnologías

El presente desarrollo se basó íntegramente en el patrón **Modelo Vista Controlador**, el cual es un patrón de arquitectura de software que divide una aplicación en tres componentes interconectados: el Modelo (lógica de datos y de negocio), la Vista (interfaz de usuario) y el Controlador (enruta las solicitudes del usuario y coordina las interacciones entre el modelo y la vista). El objetivo principal del MVC es separar las responsabilidades, lo que facilita el mantenimiento, la reutilización del código y la escalabilidad de las aplicaciones.

Pero también se usaron otros patrones como:

- **Alta cohesión:** Cada módulo tiene responsabilidades claras y bien definidas.
- **Capas:** La aplicación se organiza en niveles (presentación, lógica y datos).
- **DAO (Data Access Object):** Facilita el acceso y manipulación de datos aislando la lógica de negocio (capa de repository).
- **Modelo, Vista y Controlador (MVC):** Separa la lógica, la interfaz y el control de flujo.
- **Bajo acoplamiento:** Los componentes dependen lo menos posible entre sí.
- **Experto en información:** Cada clase maneja los datos que mejor conoce.
- **DTO (Data Transfer Object):** Objeto usado para transferir datos entre capas sin lógica adicional.

En cuanto a las tecnologías utilizadas se podrían resumir en:

- Base de Datos: MySQL
- Herramienta de Comunicación del Equipo: Discord
- Herramienta de prototipado Interfaces Gráficas UI: Balsamic
- Herramienta de Modelado: UMLetino
- Herramienta de Codificación: IntelliJ java jdk 21, spring, lombok, thymeleaf, y librerías de Mercado Pago.
- Herramienta de Gestión de Proyecto: Github y Git

3 Desarrollo de Software y dominio del problema

3.1 Especificación del Proyecto

La consigna del modelo del negocio fue:

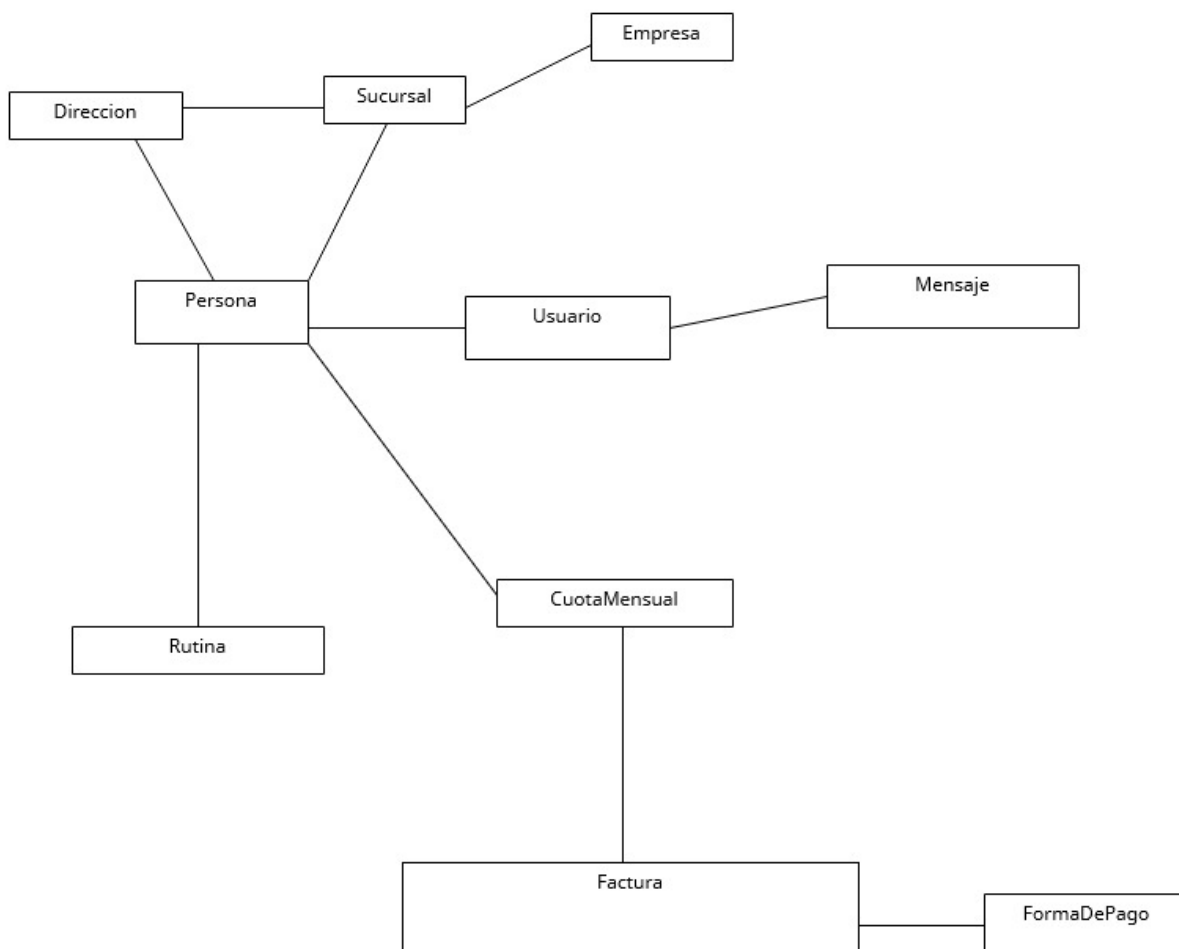
El equipo de desarrollo fue contratado por el Gimnasio “Sport” para realizar un Software que permita administrar la gestión de asociados/as que el mismo posee. Entre las funcionalidades que el software debe contemplar se encuentra: A) Alta, Consulta, Modificación y Baja de: socios/as y usuarios del sistema con sus respectivos roles. B) Gestión de cobro de cuota mensual. C)

Gestión de Deuda. D) Generación y envío de compañías promocionales por correo electrónico. E) Saludos por el cumpleaños de asociados/as. F) Gestión y seguimiento de rutina entregadas por los profesores de educación física a los socios/as.

El sistema deberá contar con tres tipos de perfiles que habilitarán las funcionalidades según los permisos que tengan cada rol de los usuarios en el sistema. A) Usuario Administrador: podrá acceder a todas las funcionalidades del sistema. B) Empleado: podrá acceder a todas las funcionalidades exceptuando Alta, Consulta, Modificación de Usuario y Alta del valor de la cuota mensual. C) Asociado/a: podrá acceder a la gestión de rutina y pago de cuota mensual para el caso que se realice con Mercado Pago. Además, podrá acceder al informe de deuda donde se detalla el total de la deuda y los meses adeudados.

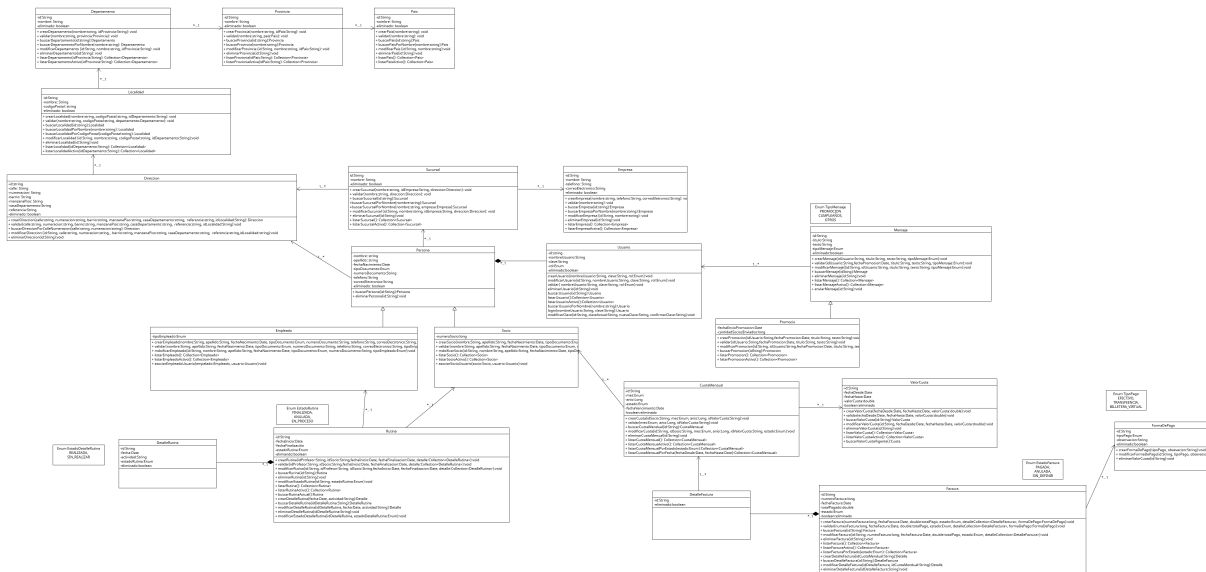
3.2 Modelado del negocio

En esta sección se presenta el diagrama de clases del **dominio**, ya que es una buena aproximación inicial para describir el problema.



3.3 Diagrama de clases del diseño

En esta sección se presenta el Diagrama de clases del **diseño**, el cual fue consensuado en clase.



3.4 Clasificación de casos de uso

Los casos de uso que se clasifican como críticos, medios y bajos.

CASOS DE USO			
GRUPO	Críticos	Medio	Bajo
1	ABM Valor Cuota y ABM Cuota Mensual	ABM Empleados	Login
2	Pago de Cuota con su generación de factura y forma de pgo.	ABM Socios	
3	ABM Rutina	ABM Sucursal	
4	Mostrar Detalle de Deuda y Mostrar todas las facturas pagadas de un mes		
5	Completar Detalle Rutinas Realizadas, Mostar Rutinas incumplidas y mostrar porcentaje cumplido de Rutina		
6	Envío Deuda, Saludos de Cumpleaños y Cuota Mensual por Correo		

A nuestro grupo le fue asignado el caso crítico de *"mostrar detalle de deuda"* y *"mostrar todas las facturas pagadas en un mes."*

3.5 Requisitos funcionales y no funcionales

Requisitos funcionales:

- El sistema debe permitir el registro de nuevos usuarios, baja y modificación.
- El sistema debe gestionar el cobro de cuotas mensuales y generación de facturas.
- El sistema debe permitir la alta, baja y actualización de datos de socios/as.
- El sistema debe mostrar las debidas rutinas para cada socio, sus facturas y deuda.
- El sistema debe poder mandar correos electrónicos de cumpleaños y promociones.
- El sistema debe permitir diferentes vistas para los diferentes roles.
- El sistema debe permitir el alta, modificación y baja de rutinas.

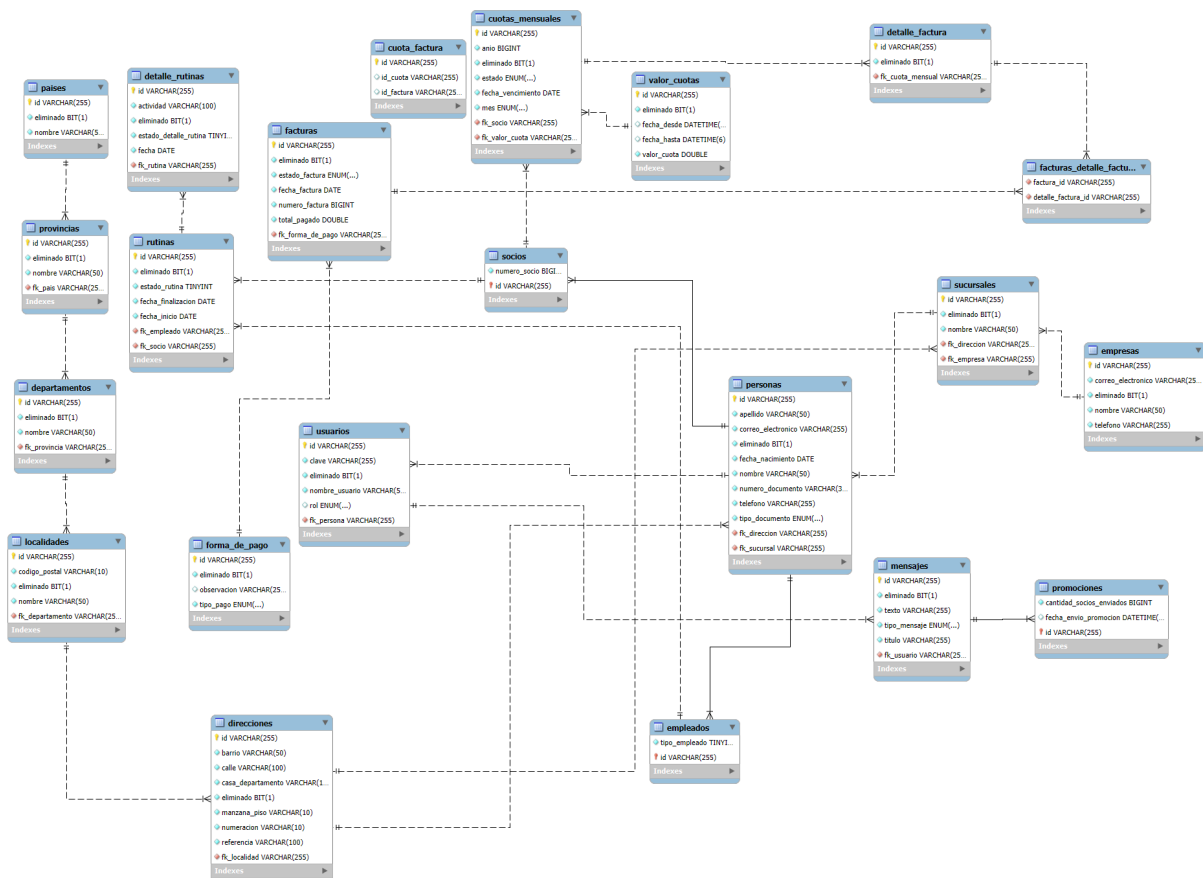
Requisitos no funcionales:

- El sistema debe garantizar disponibilidad.

- La respuesta a las operaciones críticas debe ser rápida.
- La interfaz debe ser intuitiva y fácil de usar.
- Los datos deben almacenarse de manera segura cumpliendo con normas de privacidad.
- El sistema debe ser escalable para soportar un mayor número de usuarios en el futuro.

3.6 Diagrama entidad relación

Se presenta el siguiente DER, el cual es de utilidad para ver cómo se relacionan las tablas en nuestro problema.



3.7 Prototipado

Los prototipos de pantallas son:

Google Chrome

← → ↻ <https://sportgym.com/inicio>

Sport Gym

Login

Clases

Horarios

Profesores

Google Chrome

← → ↻ <https://sportgym.com/login>

Correo electronico

Contraseña

Ingresar

Google Chrome

← → ↻ <https://sportgym.com/socio/cuotas>

Cuotas

Mes	Año	Estado	Fecha Vencimiento	Valor	Accion
Diciembre	2024	Pagada	10/12/2024	100	Pagada
Enero	2025	Pagada	10/01/2025	100	Pagada
Febrero	2025	Pagada	10/02/2025	120	Pagada
Marzo	2025	Adeudada	10/03/2025	120	Pagar
Abril	2025	Adeudada	10/04/2025	140	Pagar

Google Chrome

← → ↻ <https://sportgym.com/socio/cuotas>

Deuda

Mes	Año	Estado	Fecha Vencimiento	Valor	Accion
Marzo	2025	Adeudada	10/03/2025	120	Pagar
Abril	2025	Adeudada	10/04/2025	140	Pagar

Google Chrome
← → ↻ <https://sportgym.com/socio/rutinas/detalles>

Detalle Rutina

Fecha	Actividad	Estado
20/12/2024	Remo 3x35	Sin Realizar
20/12/2024	Press banco 2x20	Sin Realizar
20/12/2024	Bicep curl 3x25	Sin Realizar
20/12/2024	Peso muerto 3x15	Sin Realizar

Google Chrome
← → ↻ <https://sportgym.com/socio/rutinas>

Rutinas

Fecha Inicio	Fecha Fin	Estado	Profesor	Detalle
20/12/2024	20/01/2025	En Proceso	Martina Nahman	Detalle

Google Chrome
← → ↻ <https://sportgym.com/socio/inicio>

Bienvenido {Socio}

[Rutinas](#)

[Cuotas](#)

[Deudas](#)

[Cerrar Sesión](#)

Google Chrome
← → ↻ <https://sportgym.com/admin/cuotas>

Cuotas

[Ver Valor Cuota](#) [Cambiar Valor Cuota](#) [Crear Nueva Cuota](#)

Google Chrome

← → ↻ <https://sportgym.com/admin/cuotas/crearcuota>

Crear Cuota

Mes

Año

Google Chrome

← → ↻ <https://sportgym.com/admin/cuotas/valorcuota>

Gestión de Cuotas

Fecha Inicio	Fecha Fin	Valor	Estado
10/01/2025	09/03/2025	40000	Inactivo
10/03/2025	-	50000	Activo

Google Chrome

← → ↻ <https://sportgym.com/admin/cuotas/cambiarcuota>

Cambiar Valor Cuota

Valor Cuota

Google Chrome

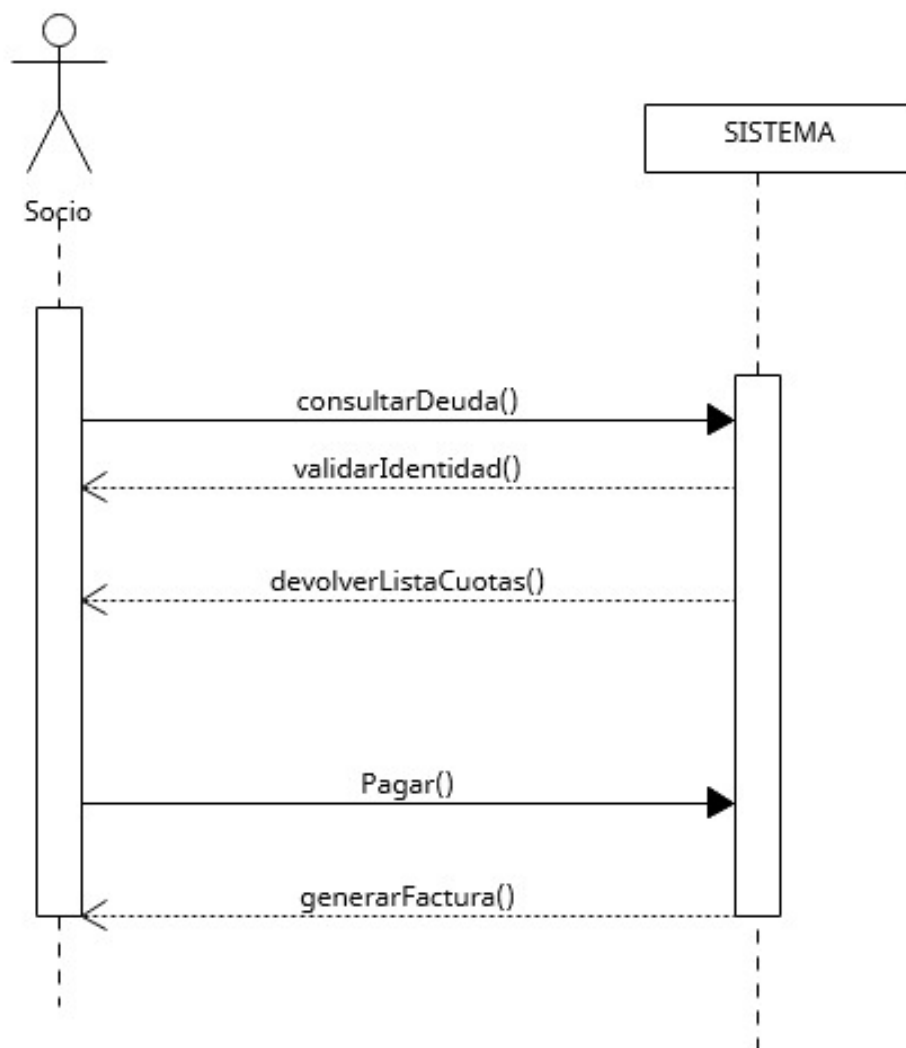
← → ↻ <https://sportgym.com/admin/deudas>

Socios con Deudas

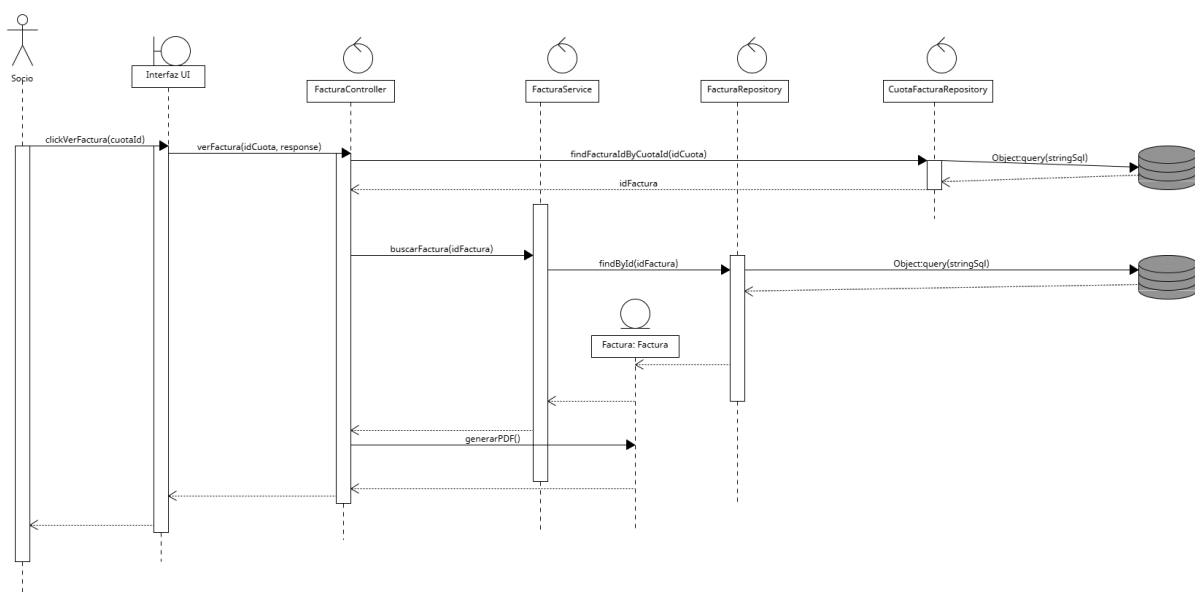
Nombre y Apellido	Meses que Adeuda	Total Deuda
Enzo Palau	2	90000
Paula Martinez	5	230000

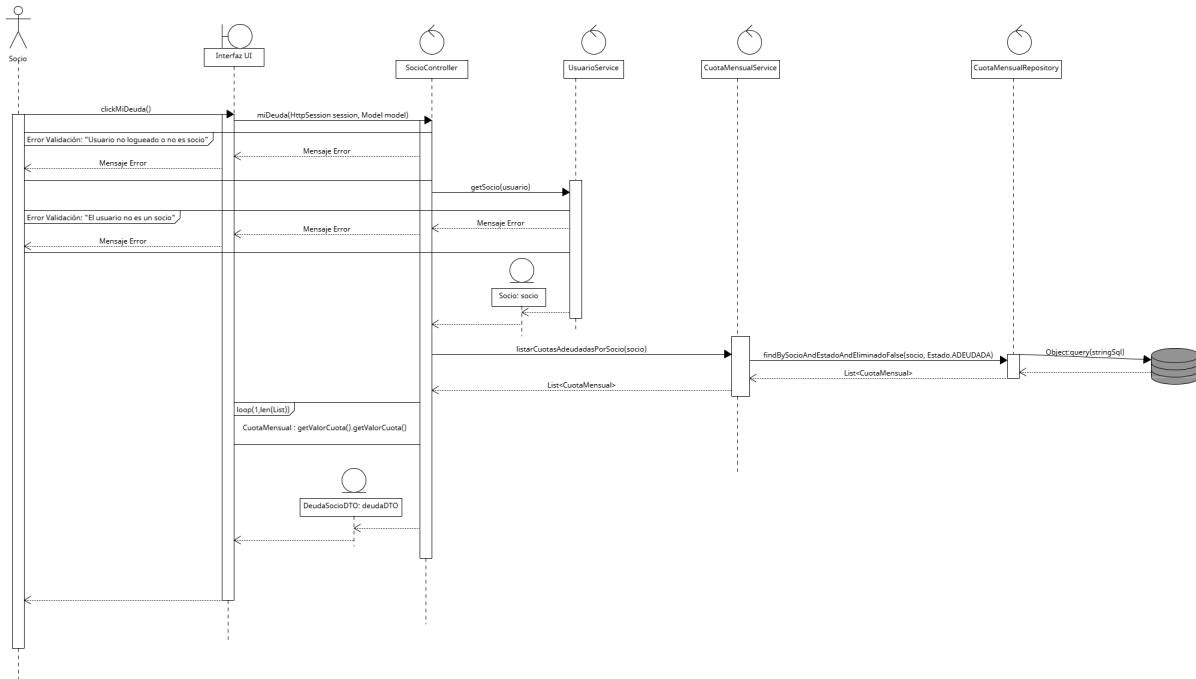
3.8 Diagramas de secuencia

Diagramas de secuencia del dominio :



Diagramas de secuencia del diseño :





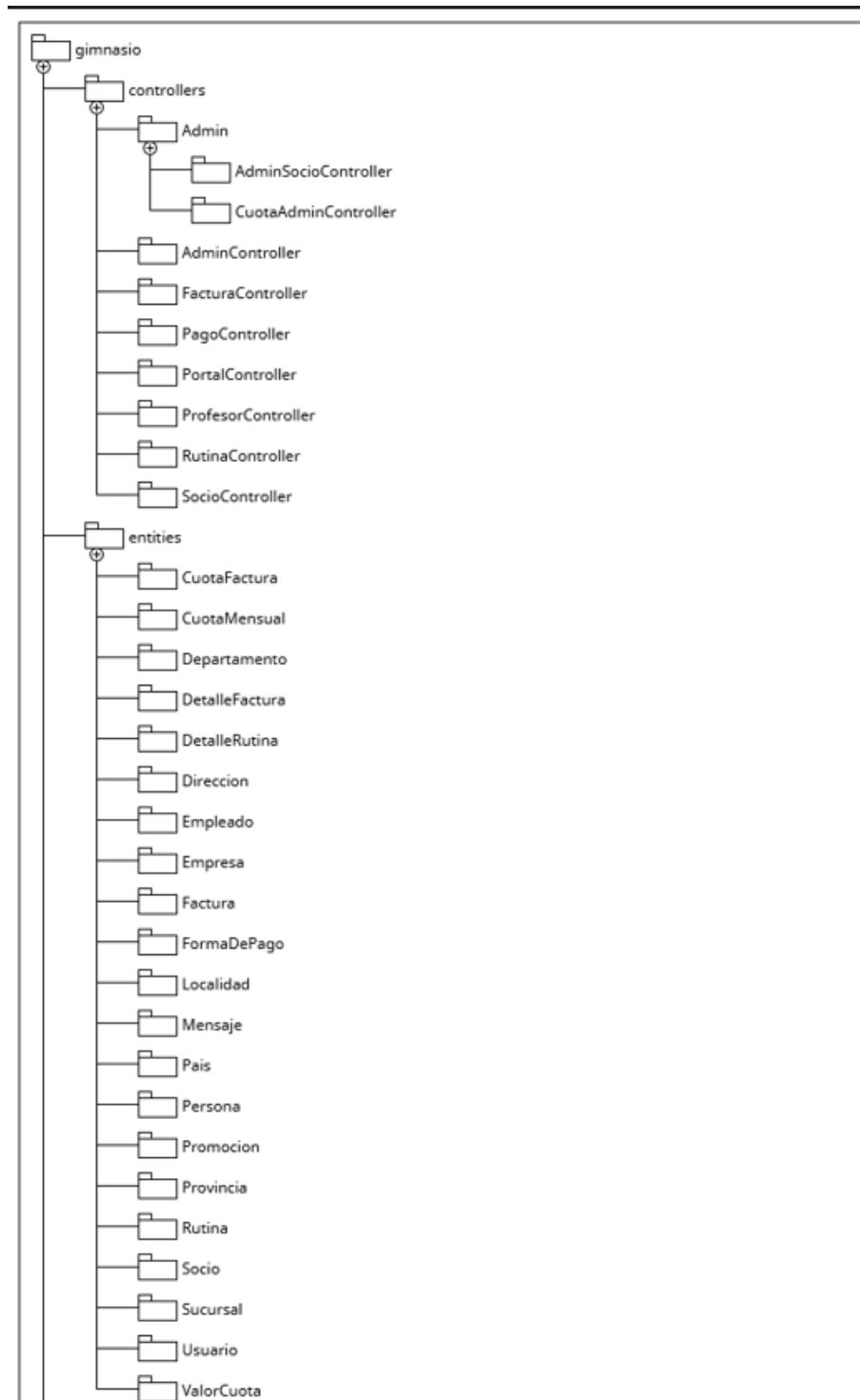
3.9 Escenarios de caso de uso

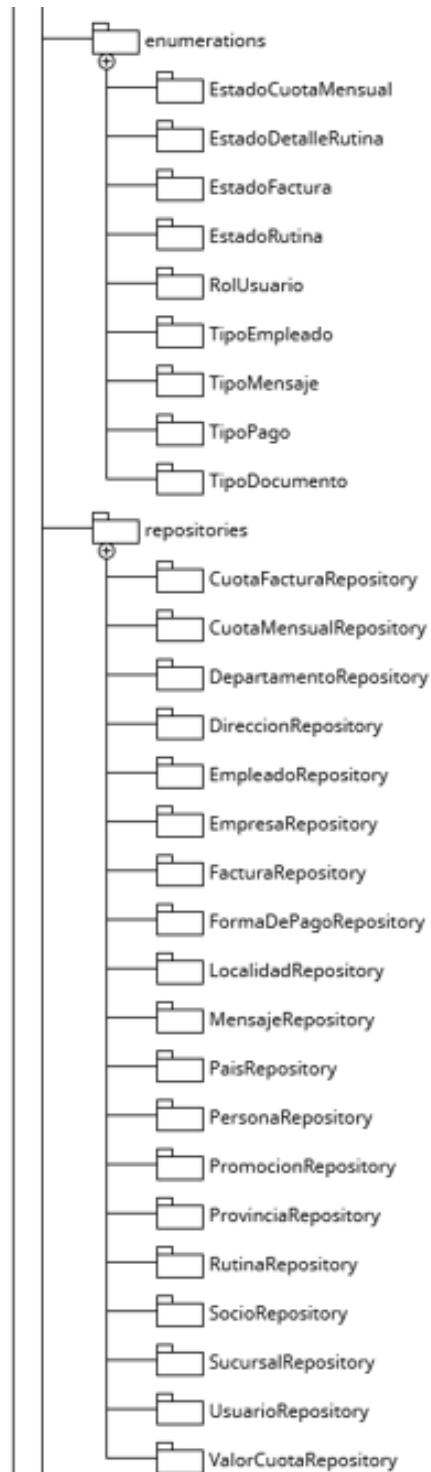
El escenario de caso de uso sobre el caso de uso crítico asignado es:

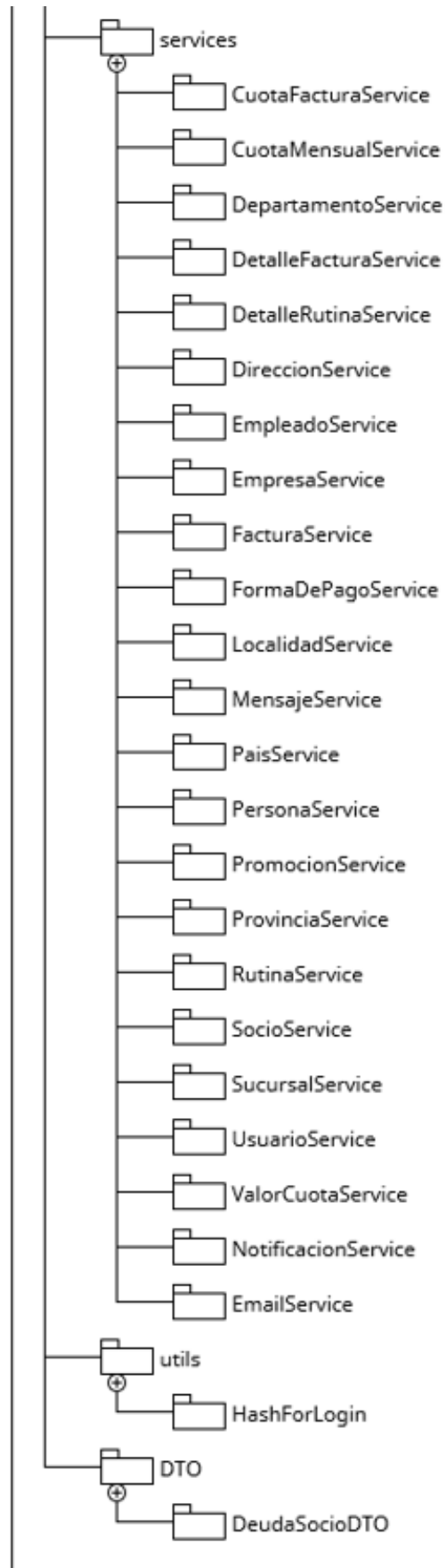
Caso de Uso	Mostrar Facturas Pagadas		ID:	
Prioridad	Media	Estimación	3 horas	ID Pantalla Prototipo: 10
Precondición	El usuario debe estar <u>logueado</u> y ser socio.			
Descripción	Permite al usuario visualizar todas las facturas pagadas			
Escenario P1	Paso	Acción		
	1	El usuario selecciona la opción "IR" en la pestaña Mis Cuotas en el menú principal del Socio		
	2	El sistema consulta todas las facturas pagadas		
	3	El sistema muestra una tabla con los meses pagados, el estado de cada cuota, la fecha de vencimiento de cada cuota, el valor de cada cuota y ver el detalle de las facturas		
	4	El usuario selecciona la opción "Ver Factura"		
Poscondición	El usuario visualizó todas las facturas pagadas.			
Excepciones	2.1	El sistema muestra un mensaje de error si hay un error de conexión a la base de datos.		
Comentarios	En la misma pestaña puede usarse la funcionalidad para pagar cuotas que <u>estén</u> adeudadas con Mercado Pago.			

Caso de Uso	Mostrar Detalle de Deuda			ID:	
Prioridad	Alta	Estimación	2 horas	ID Pantalla Prototipo:	11
Precondición	El usuario debe estar logueado y ser socio.				
Descripción	Permite al asociado visualizar el detalle de sus deudas, incluyendo los meses adeudados y el monto total pendiente.				
Escenario Pr	Paso	Acción			
	1	El usuario selecciona la opción "IR" en la pestaña Mi Deuda en el menú principal del Socio			
	2	El sistema consulta en la base de datos las deudas pendientes del socio			
	3	El sistema muestra una tabla con los meses adeudados, el valor de cada mes adeudado, el valor total de deuda y la opción "Pagar"			
Poscondición	El socio visualizó sus deudas y el detalle de cada factura.				
Excepciones	2.1	El sistema muestra un mensaje de error si hay un error de conexión a la base de datos.			
Comentario	Puede usarse la funcionalidad para pagar deudas desde la misma pantalla con Mercado Pago.				

3.10 Diagrama de Paquetes







4 Propuesta de Mejora

La propuesta se basa en que una persona pueda tener doble rol, por ejemplo, como pasa en la facultad, que si un profe realiza un curso, no podrá acceder porque es ya profesor. Entonces, en nuestro caso, un profesor puede ser a la misma vez socio.

Usuario
<pre>-id:string -nombreUsuario:String -clave:String -rol:List<RolSocio> -elimnado:boolean crearUsuario(nombreUsuario:String, clave:String, rol:List<RolSocio>):void modificarUsuario(id:String, nombreUsuario:String, clave:String, rol:List<RolSocio>):void validar(nombreUsuario:String, clave:String, rol:List<RolSocio>):void eliminarUsuario(id:String):void buscarUsuario(id:String):Usuario listarUsuario():Collection<Usuario> listarUsuarioActivo():Collection<Usuario> buscarUsuarioPorNombre(nombre:string):Usuario login(nombreUsuario:String, clave:String):Usuario modificarClave(id:String, claveActual:String, nuevaClave:String, confirmarClave:String):void</pre>

En esta clase se mantienen los atributos de socios, pero este tiene una lista de roles posibles.

5 Propuesta de Refactorización

5.1 Propuesta de controladores

La principal propuesta de mejora en este aspecto es que, bajo **ningún** concepto, realizar operaciones de reglas de negocios en los controladores.

5.2 Propuesta de dominio de la aplicación

La principal propuesta viene de la mano con la propuesta anterior de centralizar completamente las reglas de negocio en los servicios.

5.3 Propuesta de acceso a datos

Cuando se paga la deuda y se genera una factura, se creó una tabla intermedia que nos facilitaba tener a disposición el id de la factura con el id de la cuota pagada. Esto podría solucionarse con una *query* que use una estrategia de *JOIN*.

6 Porcentaje de codificación

Dado que lo que faltaría en el presente proyecto es terminar de completar la vista de los ABM menos críticos, consideramos que el porcentaje de codificación es un **85** por ciento.

7 Material de Exposición

- Link a la presentación: https://www.canva.com/design/DAGz5J25vQs/Pa-me1ZS1j4m9rL66au1nQ/edit?utm_campaign=designshareutm_medium=link2utm_source=sharebutton
- Link al repositorio: <https://github.com/Paulonia14/Gimnasio-IS2>

8 Referencias

- -Material propuesto por la cátedra
- -Proyectos Tinder de mascotas y Videojuegos: <https://github.com/martupiru/IS2-Sprint>