

Anexo IV. Diseño del sistema

Sistema para la mejora de la movilidad articular basada en el uso de la estimación de posturas

Trabajo Fin de Máster

Ingeniería Informática

Febrero 2025



VNiVERSiDAD
D SALAMANCA

Autor

Sergio Salinero Santamaría

Tutores

André Fílipe Sales Mendes

Gabriel Villarrubia González

ÍNDICE

1. Introducción	1
2. Patrones arquitectónicos.....	2
2.1. Patrón de capas	2
2.2. Patrón cliente-servidor.....	3
2.3. Patrón dao.....	3
2.4. Patrón singleton	4
3. Subsistemas de diseño.....	5
4. Clases de diseño	6
4.1. Subsistema datos.....	6
4.1.1. Servicios	6
4.1.2. Controlador	6
4.1.3. Modelo	7
4.2. Subsistema backend	8
4.2.1. Servicios	8
4.2.2. Controlador	8
4.2.3. Modelo	9
4.2.4. Utilidades	9
4.3. Subsistema frontend.....	10
4.3.1. Vistas	10
4.3.2. Controlador	10
5. Realización de casos de uso	11
5.1. Gestión de autenticación	11
5.2. Gestión de usuarios	13
5.3. Gestión de rutinas	15
5.4. Gestión de ejercicios	21
5.5. Gestión de estadísticas	23
6. Diseño de la base de datos.....	24
7. Modelo de despliegue.....	25
8. Referencias.....	26

ÍNDICE DE FIGURAS

Figura 1: Patrón de capas.....	2
Figura 2: Patrón cliente-servidor	3
Figura 3: Patrón DAO	3
Figura 4: Patrón singleton.....	4
Figura 5: Subsistemas de diseño	5
Figura 6: Clase de diseño Servicios del subsistema Datos	6
Figura 7: Clase de diseño Controlador del subsistema Datos	6
Figura 8: Clase de diseño Modelo del subsistema Datos	7
Figura 9: Clase de diseño Servicios del subsistema Backend.....	8
Figura 10: Clase de diseño Controlador del subsistema Backend	8
Figura 11: Clase de diseño Modelo del subsistema backend.....	9
Figura 12: Clase de diseño Utilidades del subsistema Backend	9
Figura 13: Clases de diseño Vistas del subsistema Frontend	10
Figura 14: Clase de diseño Controlador del subsistema Frontend	10
Figura 15: Realización de caso de uso [UC-001] Iniciar sesión.....	11
Figura 16: Realización de caso de uso [UC-002] Establecer contraseña	11
Figura 17: Realización de caso de uso [UC-003] Restablecer contraseña	12
Figura 18: Realización de caso de uso [UC-004] Cerrar sesión	12
Figura 19: Realización de caso de uso [UC-005] Ver datos de perfil.....	13
Figura 20: Realización de caso de uso [UC-006] Cambiar contraseña	13
Figura 21: Realización de caso de uso [UC-007] Actualizar peso	14
Figura 22: Realización de caso de uso [UC-008] Eliminar cuenta.....	14
Figura 23: Realización de caso de uso [UC-009] Elaborar una rutina	15
Figura 24: Realización de caso de uso [UC-010] Mostrar categorías.....	15
Figura 25: Realización de caso de uso [UC-011] Publicar rutina en categoría	16
Figura 26: Realización de caso de uso [UC-012] Acceder a categoría.....	16
Figura 27: Realización de caso de uso [UC-013] Eliminar rutina de categoría	17
Figura 28: Realización de caso de uso [UC-014] Limpiar categoría.....	17
Figura 29: Realización de caso de uso [UC-015] Mostrar historial de rutinas.....	18
Figura 30: Realización de caso de uso [UC-016] Limpiar historial de rutinas	18
Figura 31: Realización de caso de uso [UC-017] Ejecutar una rutina.....	19
Figura 32: Realización de caso de uso [UC-018] Reconocer ejercicio	19
Figura 33: Realización de caso de uso [UC-019] Terminar rutina	20
Figura 34: Realización de caso de uso [UC-020] Mostrar ejercicios.....	21
Figura 35: Realización de caso de uso [UC-021] Añadir ejercicio	21
Figura 36: Realización de caso de uso [UC-022] Modificar ejercicio	22

Figura 37: Realización de caso de uso [UC-023] Eliminar ejercicio.....	22
Figura 38: Realización de caso de uso [UC-024] Mostrar estadísticas.....	23
Figura 39: Realización de caso de uso [UC-025] Limpiar estadísticas	23
Figura 40: Diseño de la base de datos.....	24
Figura 41: Diagrama de despliegue	25

1. INTRODUCCIÓN

El propósito del presente documento es describir el modelo de diseño del sistema. El desarrollo de una solución de software abarca la definición de los requisitos del sistema, la identificación de módulos y componentes, la especificación de la arquitectura, así como la definición de las estructuras de datos y algoritmos que se emplearán. También se analizará el impacto que estas decisiones tienen en el sistema.

La organización del anexo se presenta a continuación:

- **Patrones arquitectónicos:** Soluciones ampliamente aceptadas y aplicables a diversos sistemas, probadas por la experiencia en la industria.
- **Subsistemas de diseño:** Módulo o componente que conforman el sistema de software.
- **Clases de diseño:** Definición de los atributos, operaciones y relaciones de las clases de diseño del sistema.
- **Realización de casos de uso en el modelo de diseño:** Transformación de los requisitos funcionales definidos en la etapa de análisis en un diseño técnico detallado.
- **Modelo de despliegue:** Representación gráfica del entorno físico en el que se ejecuta el sistema de software.
- **Descripción de la arquitectura:** Perspectiva general de la estructura del sistema, incluyendo sus componentes, interacciones y comportamiento.

2. PATRONES ARQUITECTÓNICOS

Los patrones arquitectónicos son soluciones probadas y documentadas para problemas recurrentes en el diseño y desarrollo de sistemas software. A la hora de escoger estos patrones hay que tener en cuenta las tecnologías que se utilizarán en el desarrollo del sistema:

- **Estimación de posturas:** Técnicas de visión computacional que detectan figuras humanas en imágenes y vídeos, permitiendo determinar la posición de las articulaciones de un usuario.
- **Spring boot 4:** Tecnología diseñada para simplificar el desarrollo de la lógica de negocio de un sistema. Ofrece un enfoque modular que facilita la configuración, integración y despliegue de aplicaciones Java.
- **React:** Biblioteca JavaScript diseñada para la construcción de interfaces de usuario dinámicas y reutilizables en aplicaciones web.

2.1. PATRÓN DE CAPAS

El patrón de capas organiza la arquitectura lógica de un sistema en distintas capas, cada una con responsabilidades específicas y relacionadas, promoviendo una colaboración estructurada y un acoplamiento controlado, principalmente desde las capas superiores hacia las inferiores.

Las capas definidas por este patrón son las siguientes:

- **Capa de presentación:** Encargada de la interfaz de usuario, esta capa gestiona la visualización de información al usuario y facilita la interacción del mismo con el sistema.
- **Capa de lógica de negocio:** Corresponde al núcleo funcional de la aplicación. Aquí se establecen las reglas de negocio, se validan los datos y se gestionan procesos complejos que integran distintas operaciones.
- **Capa de datos:** Dedicada a la administración de la información, incluye la base de datos, los sistemas de almacenamiento y otros componentes relacionados con la gestión de datos.

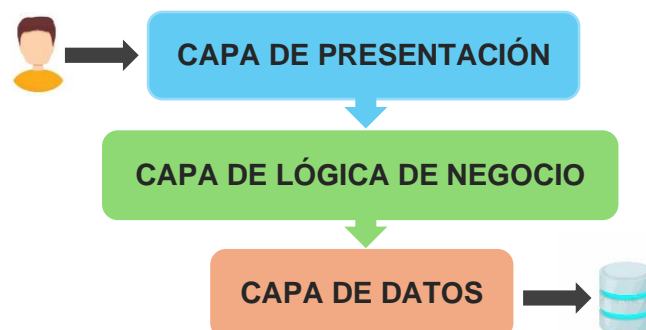


Figura 1: Patrón de capas

2.2. PATRÓN CLIENTE-SERVIDOR

El sistema se organiza en dos componentes principales: el cliente y el servidor. Estos se comunican entre sí a través de una red utilizando protocolos de comunicación como HTTP o TCP/IP. El intercambio de información se basa en un modelo de solicitud-respuesta, en el cual el cliente emite una solicitud al servidor y permanece a la espera de una respuesta.

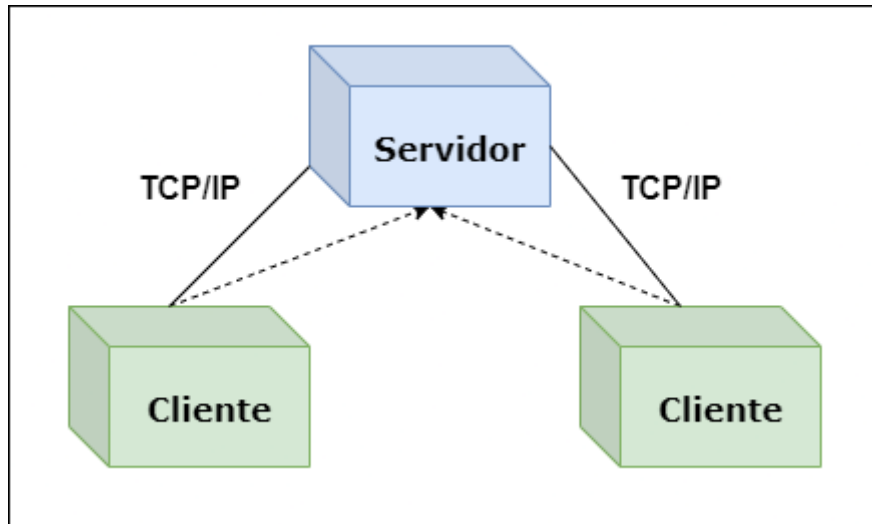


Figura 2: Patrón cliente-servidor

2.3. PATRÓN DAO

El patrón DAO (Data Access Object) facilita la separación entre la lógica de negocio y la lógica de acceso a los datos. Este enfoque introduce una capa de abstracción que actúa como intermediaria entre la capa de lógica de negocio y la capa de datos, garantizando que las modificaciones realizadas en una de estas capas no afecten a la otra.

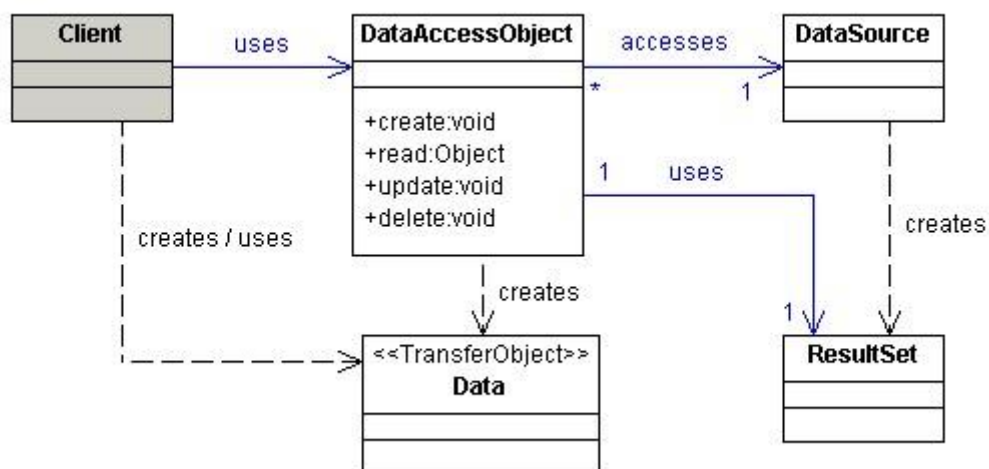


Figura 3: Patrón DAO

2.4. PATRÓN SINGLETON

Se busca garantizar que una clase posea una única instancia, accesible de manera global y concreta. Este patrón será aplicado en el servidor, el cual debe ser único y actuar como un punto centralizado para todos los clientes.

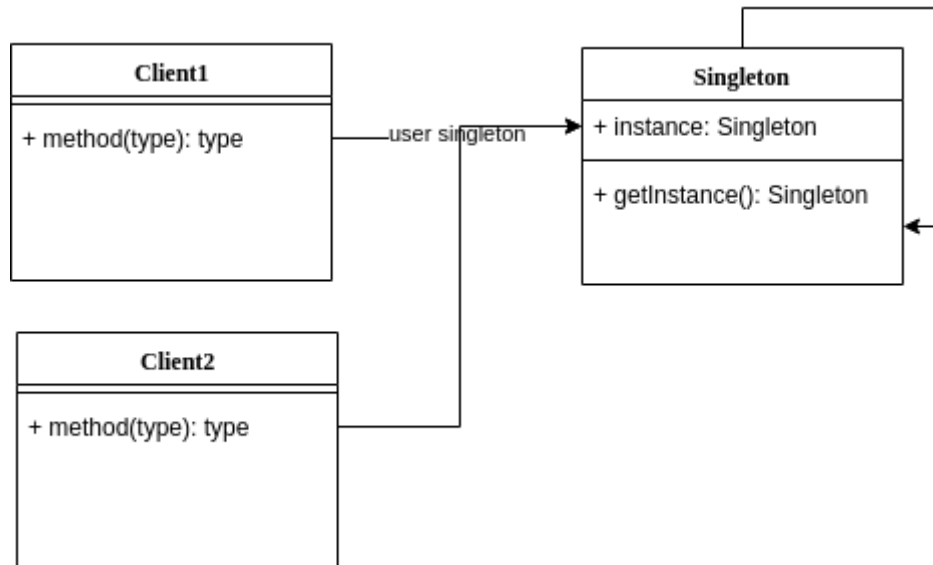


Figura 4: Patrón singleton

3. SUBSISTEMAS DE DISEÑO

En este apartado se presentan los subsistemas de diseño correspondientes a los distintos módulos que conforman el sistema. Se detalla la funcionalidad asociada a cada componente, poniendo énfasis en su diseño para garantizar una mayor modularidad y escalabilidad del sistema, además de facilitar su mantenimiento y futura evolución.

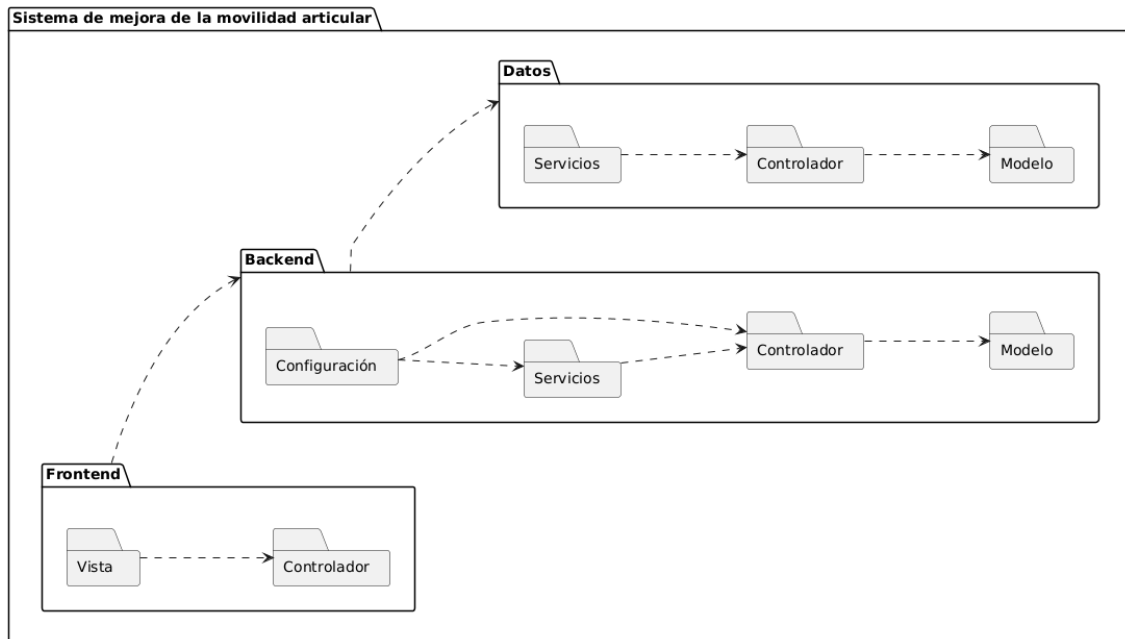


Figura 5: Subsistemas de diseño

En la *Figura 5* se observan los subsistemas de diseño presentes en el sistema, dividiéndose en los siguientes módulos:

- **Frontend:** Alberga la parte visible del sistema, representando el conjunto de acciones con las que el usuario final podrá interactuar de forma directa.
- **Backend:** Encargado de gestionar la lógica de negocio, interconecta los componentes *Frontend* y *Datos*.
- **Datos:** Comprende un servidor cuyo fin es gestionar el acceso a los datos, así como operar sobre ellos.

4. CLASES DE DISEÑO

Las clases de diseño constituyen una representación abstracta de una o más clases utilizadas en la implementación del sistema. A continuación, se detallan las clases de diseño que integran cada uno de los subsistemas previamente definidos.

4.1. SUBSISTEMA DATOS

4.1.1. Servicios

Conjunto de servicios disponibles para componentes backend dentro del sistema.

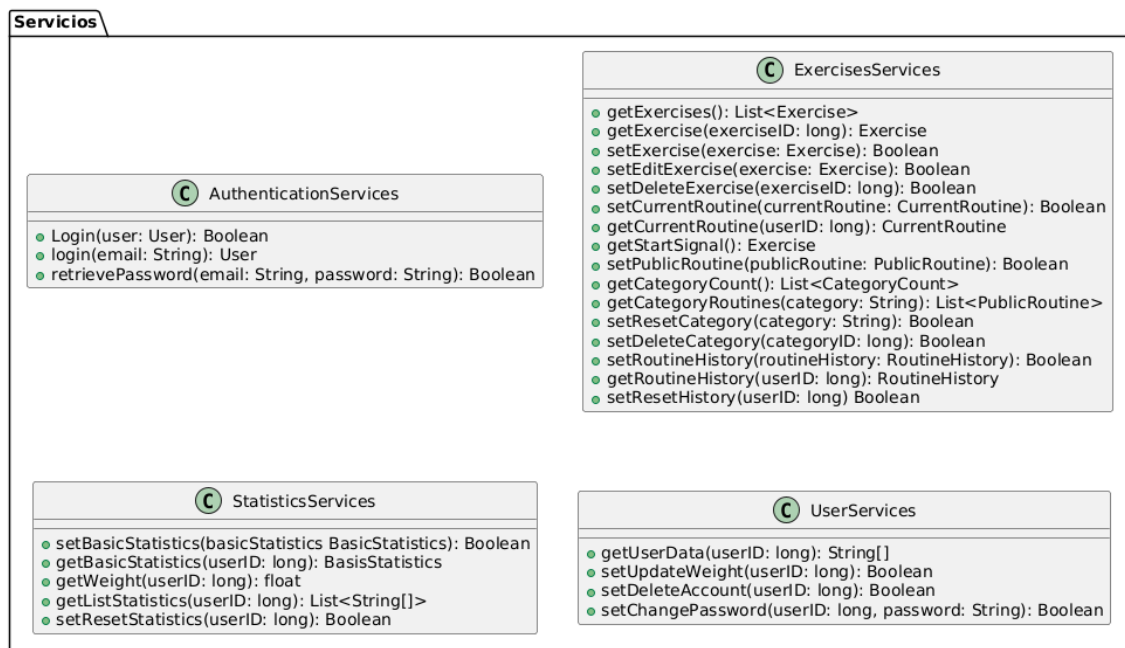


Figura 6: Clase de diseño Servicios del subsistema Datos

4.1.2. Controlador

Lógica y operaciones internas para gestionar el acceso a la base de datos una vez se ha solicitado un servicio.

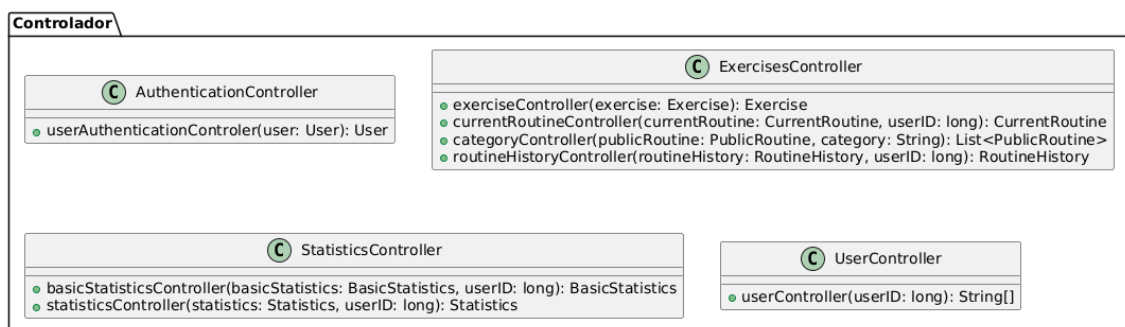


Figura 7: Clase de diseño Controlador del subsistema Datos

4.1.3. Modelo

Almacenamiento temporal de conjuntos de datos que serán utilizados en las operaciones con la base de datos.



Figura 8: Clase de diseño Modelo del subsistema Datos

4.2. SUBSISTEMA BACKEND

4.2.1. Servicios

Conjunto de servicios disponibles para componentes frontend dentro del sistema.

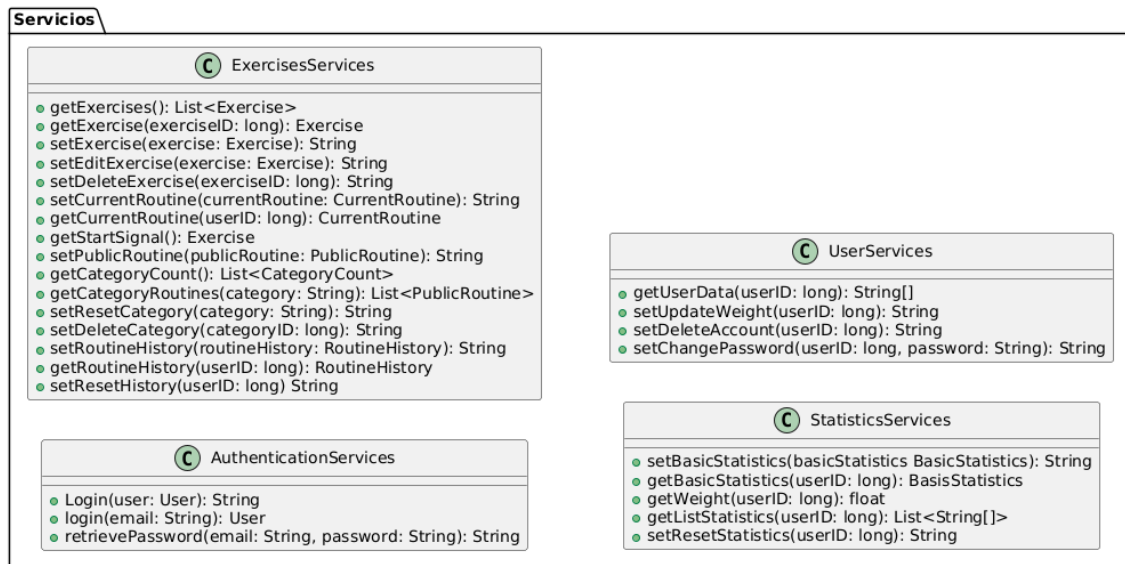


Figura 9: Clase de diseño Servicios del subsistema Backend

4.2.2. Controlador

Controlador de la lógica para gestionar la información recibida/solicitada por componentes software, estará interconectado con el servidor de base de datos.

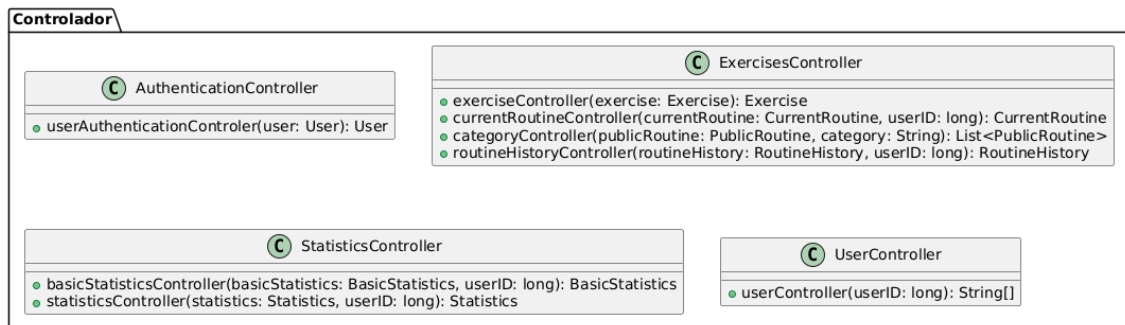


Figura 10: Clase de diseño Controlador del subsistema Backend

4.2.3. Modelo

Almacenamiento temporal de conjuntos de datos utilizados en la gestión de las peticiones a los servidores.

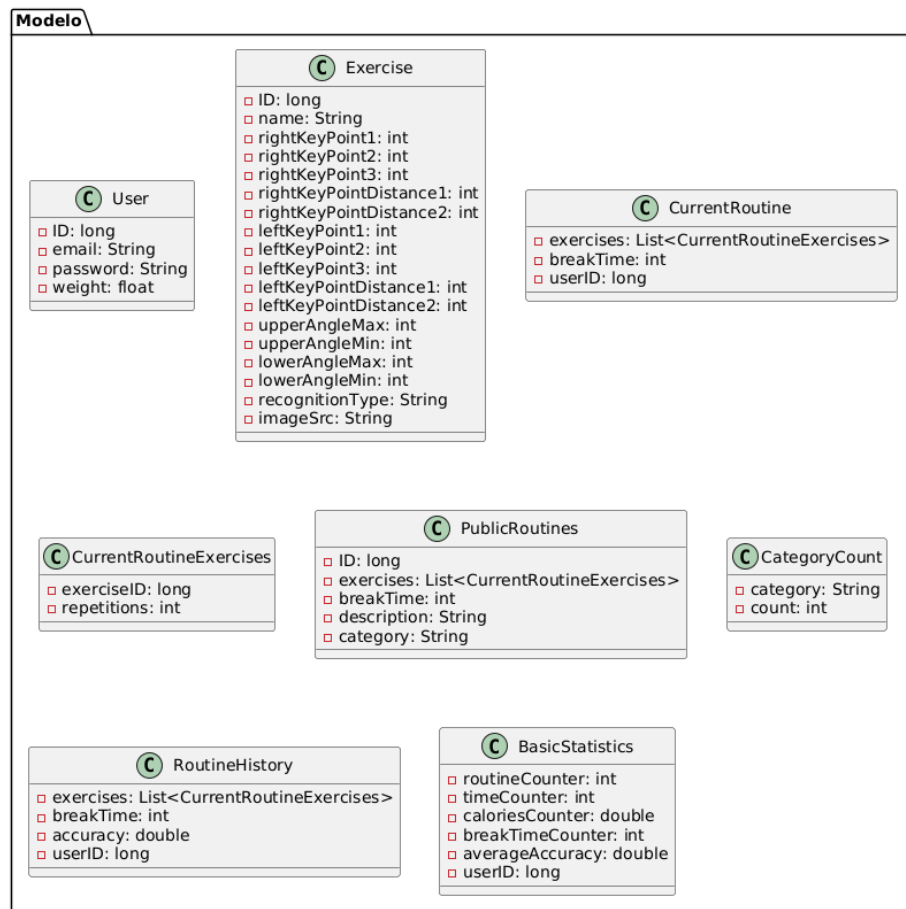


Figura 11: Clase de diseño Modelo del subsistema Backend

4.2.4. Utilidades

Conjunto de clases de configuración dedicadas a dar soporte al backend en operaciones internas.

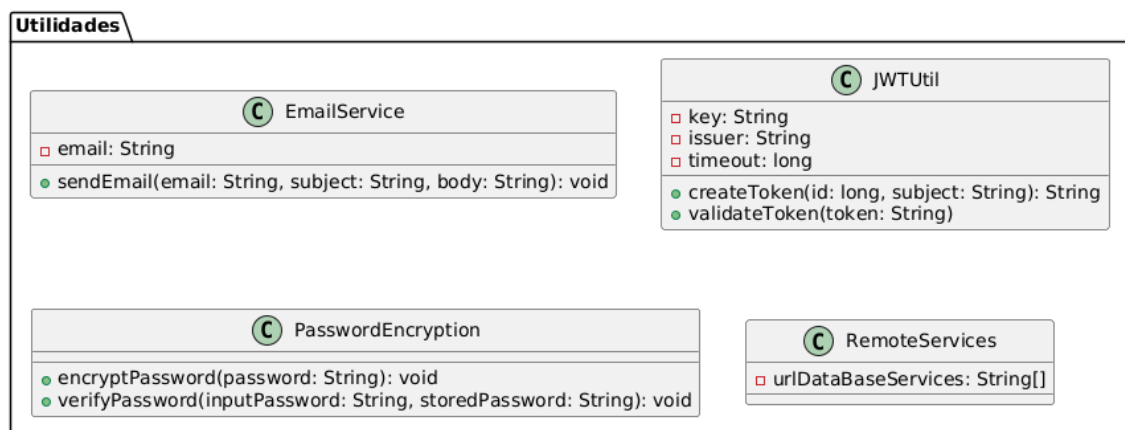


Figura 12: Clase de diseño Utilidades del subsistema Backend

4.3. SUBSISTEMA FRONTEND

4.3.1. Vistas

Conjunto de clases que conforman la interfaz de usuario.

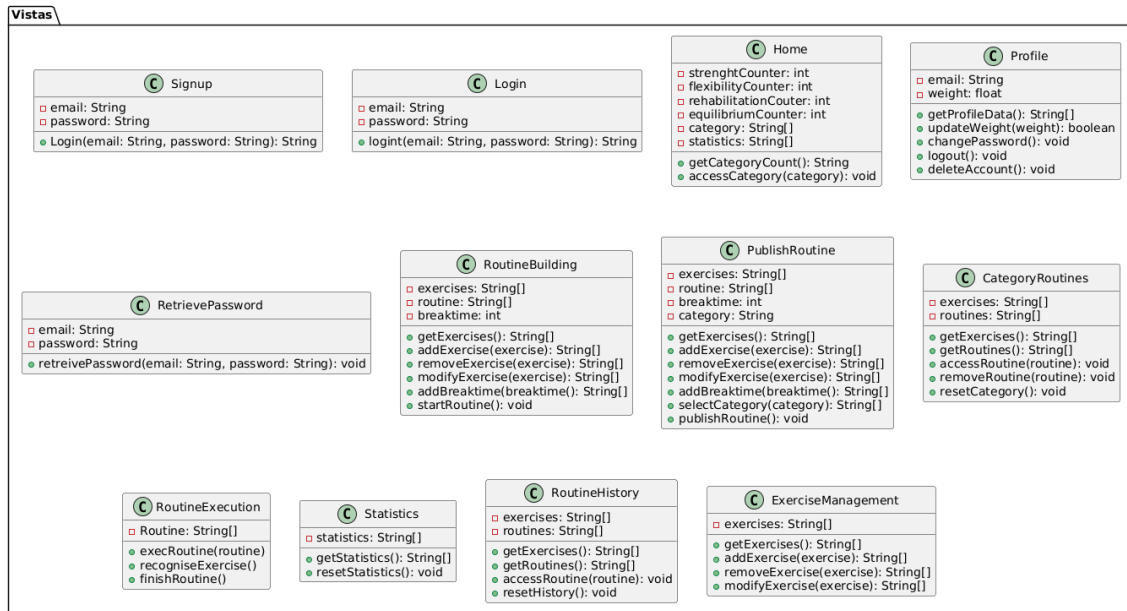


Figura 13: Clases de diseño Vistas del subsistema Frontend

4.3.2. Controlador

Lógica y operaciones de comunicación con el servidor backend dependiente de las acciones del usuario.

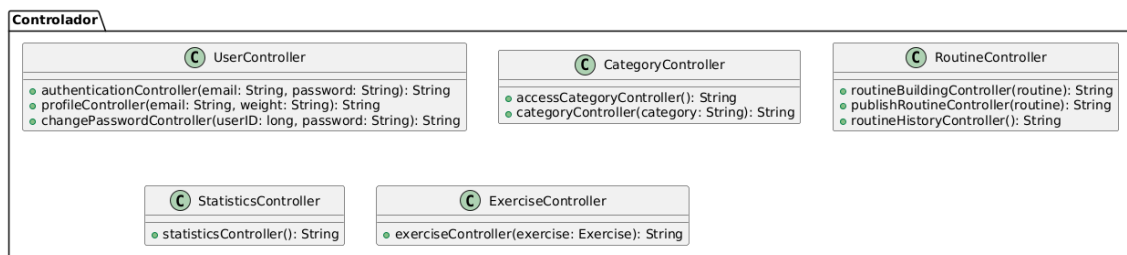


Figura 14: Clase de diseño Controlador del subsistema Frontend

5. REALIZACIÓN DE CASOS DE USO

En esta sección se describirá de manera detallada la forma en que se llevará a cabo la implementación de cada uno de los casos de uso identificados durante la etapa de recopilación de requisitos y análisis del sistema.

5.1. GESTIÓN DE AUTENTICACIÓN

- [UC-001] Iniciar sesión

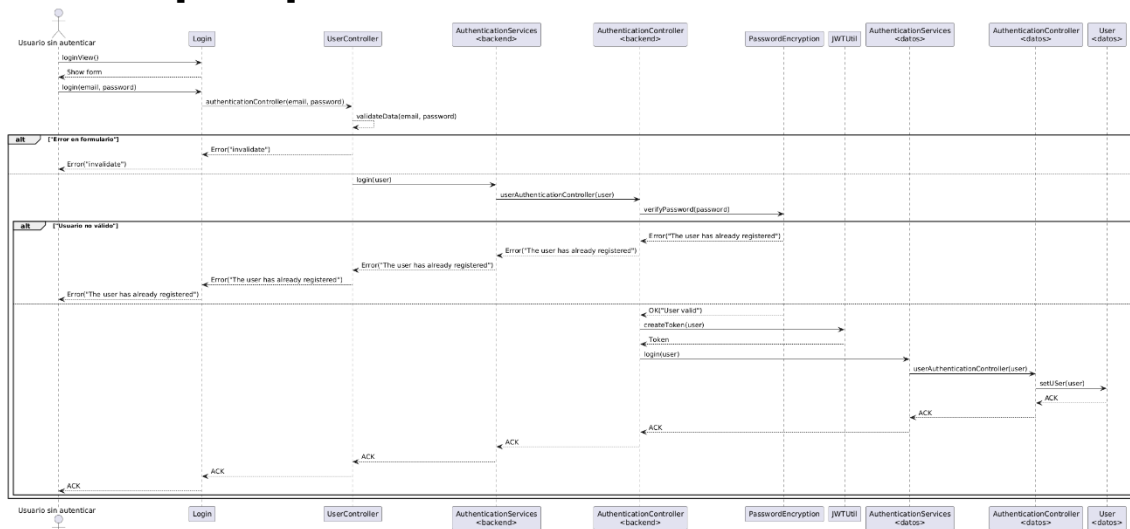


Figura 15: Realización de caso de uso [UC-001] Iniciar sesión

- [UC-002] Establecer contraseña

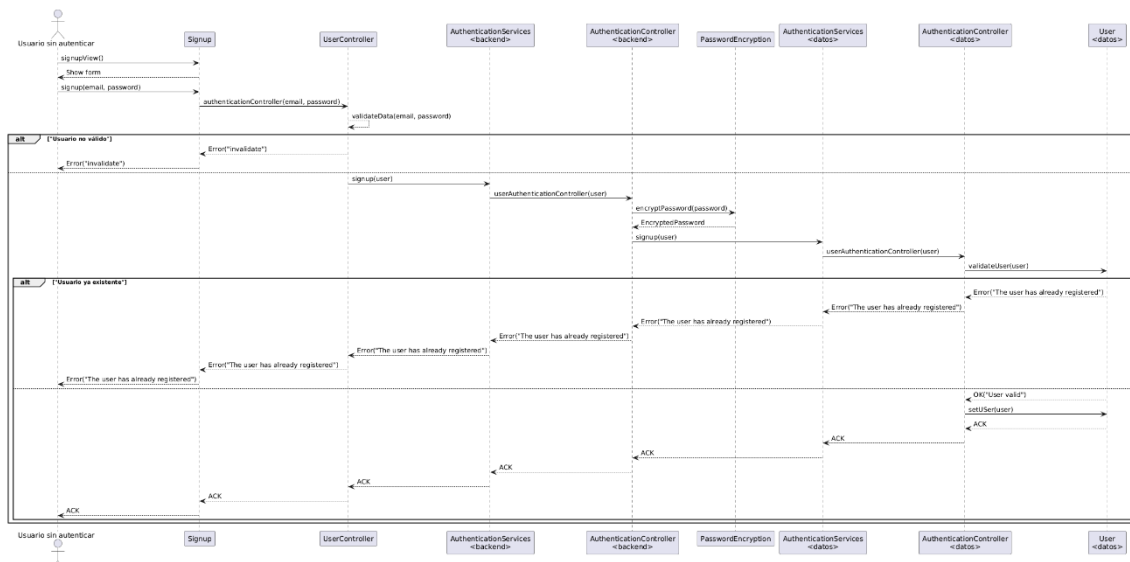


Figura 16: Realización de caso de uso [UC-002] Establecer contraseña

- **[UC-003] Restablecer contraseña**

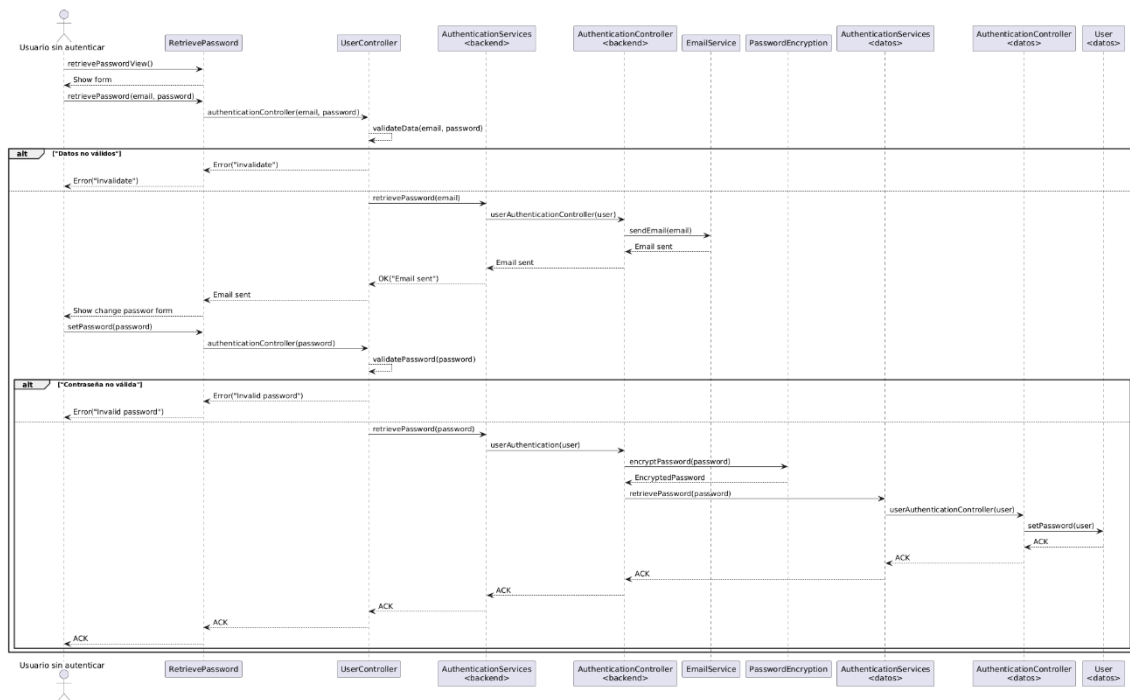


Figura 17: Realización de caso de uso [UC-003] Restablecer contraseña

- **[UC-004] Cerrar sesión**

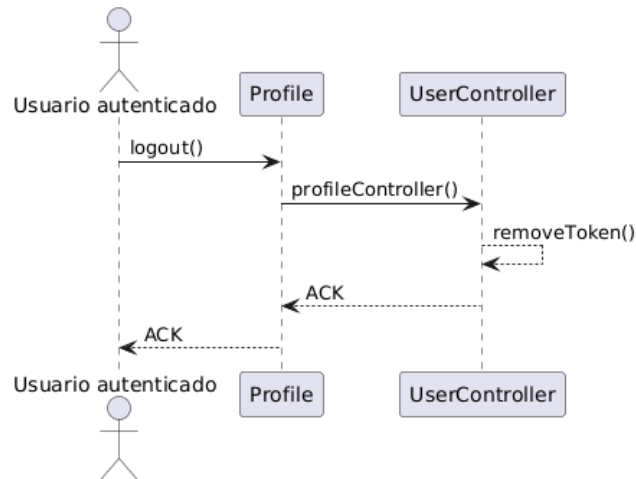


Figura 18: Realización de caso de uso [UC-004] Cerrar sesión

5.2. GESTIÓN DE USUARIOS

- [UC-005] Ver datos de perfil

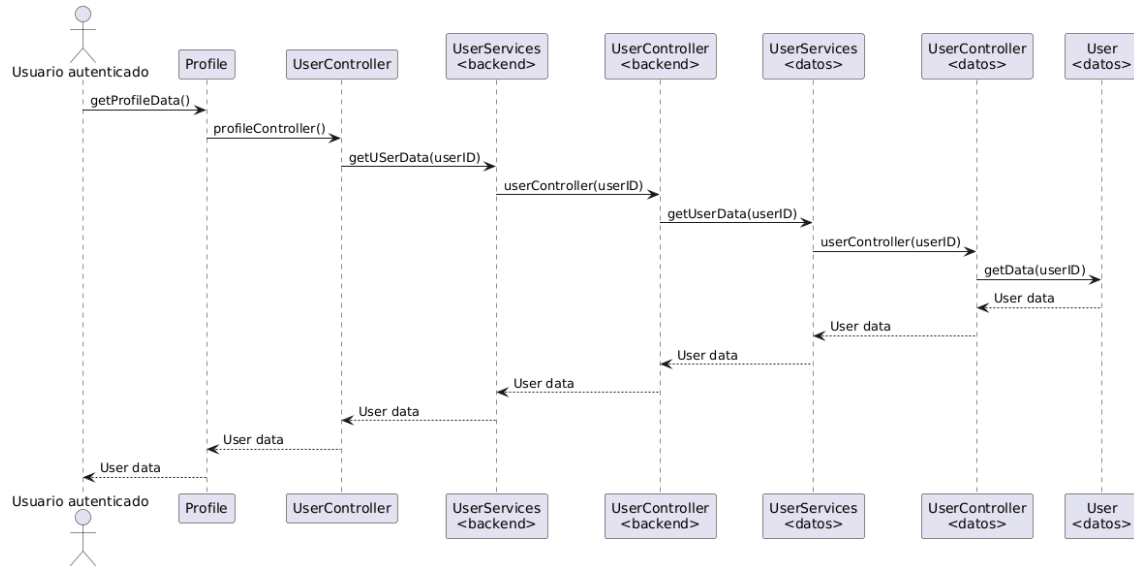


Figura 19: Realización de caso de uso [UC-005] Ver datos de perfil

- [UC-006] Cambiar contraseña

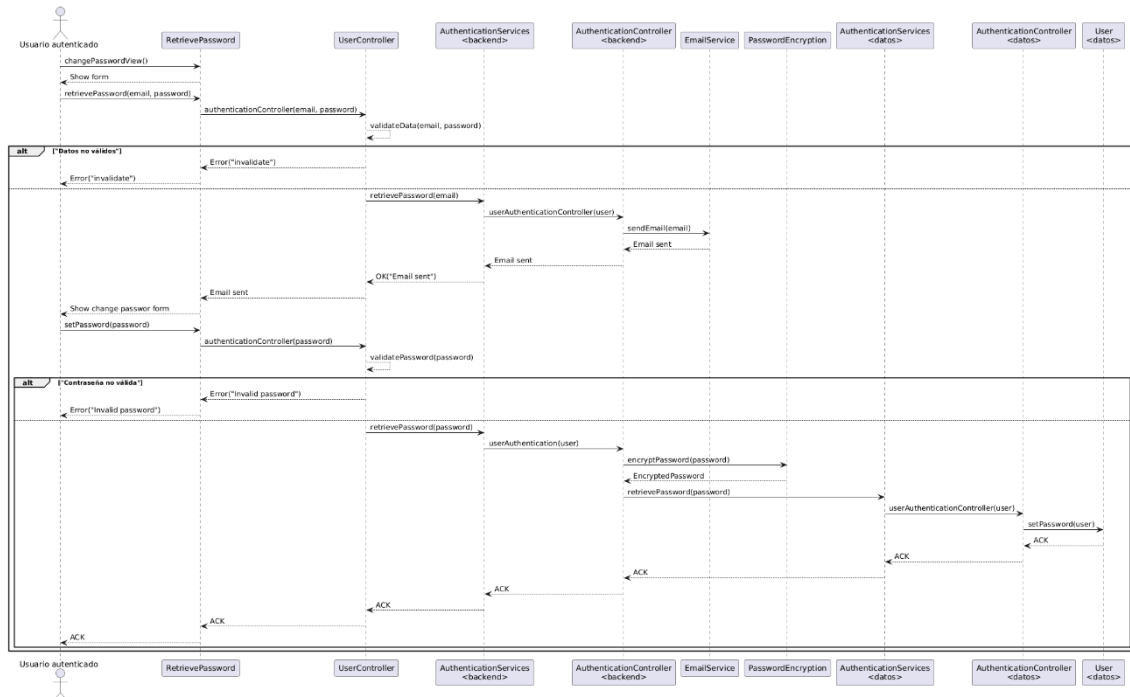


Figura 20: Realización de caso de uso [UC-006] Cambiar contraseña

- **[UC-007] Actualizar peso**

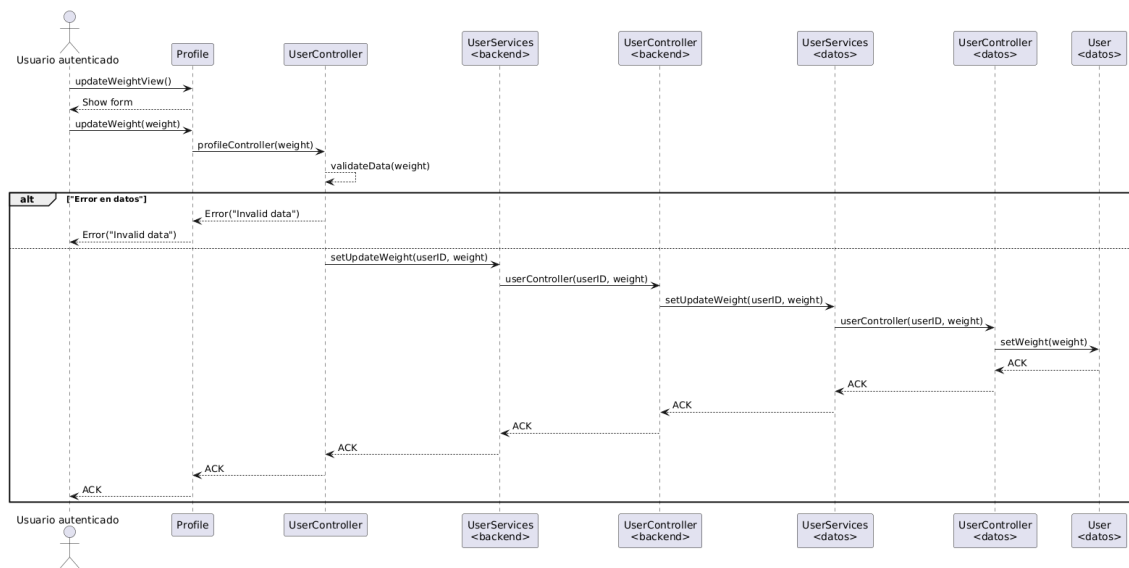


Figura 21: Realización de caso de uso [UC-007] Actualizar peso

- **[UC-008] Eliminar cuenta**

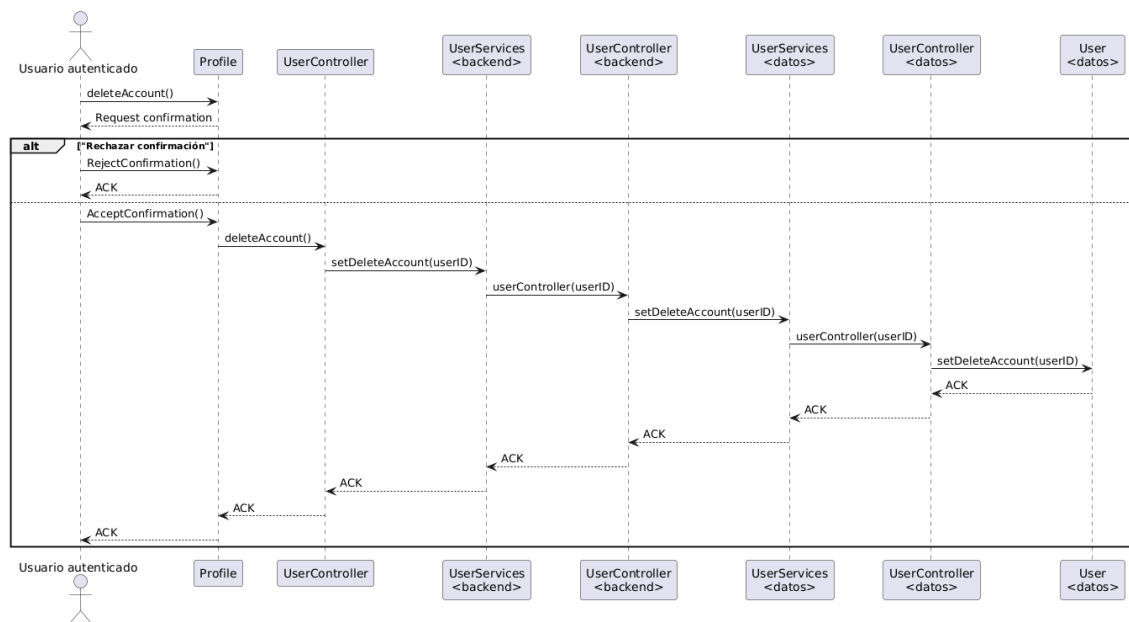


Figura 22: Realización de caso de uso [UC-008] Eliminar cuenta

5.3. GESTIÓN DE RUTINAS

- [UC-009] Elaborar una rutina

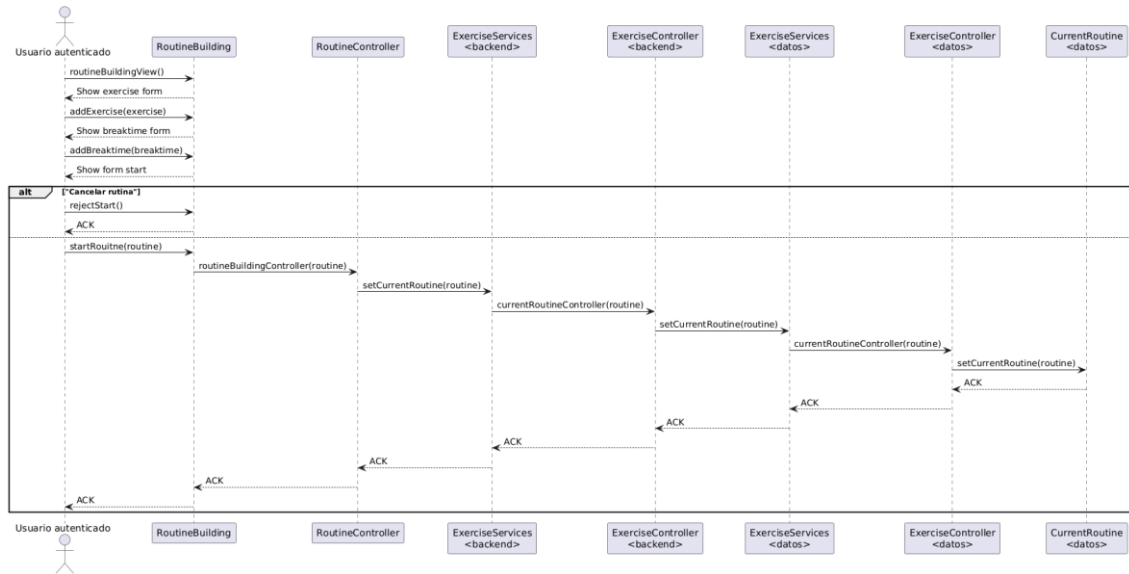


Figura 23: Realización de caso de uso [UC-009] Elaborar una rutina

- [UC-010] Mostrar categorías

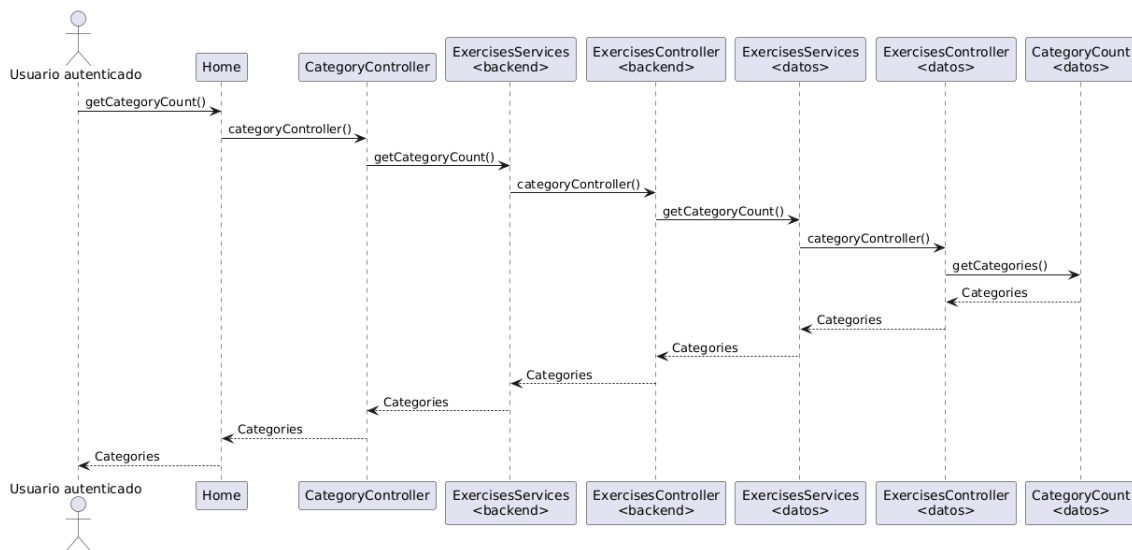


Figura 24: Realización de caso de uso [UC-010] Mostrar categorías

- [UC-011] Publicar rutina en categoría

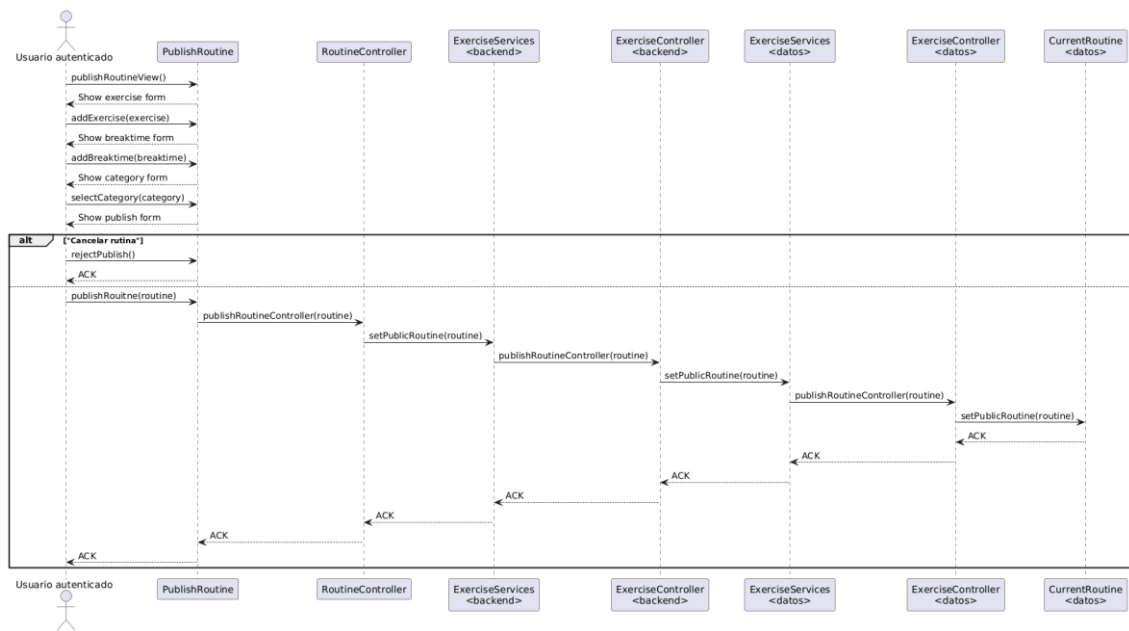


Figura 25: Realización de caso de uso [UC-011] Publicar rutina en categoría

- [UC-012] Acceder a categoría

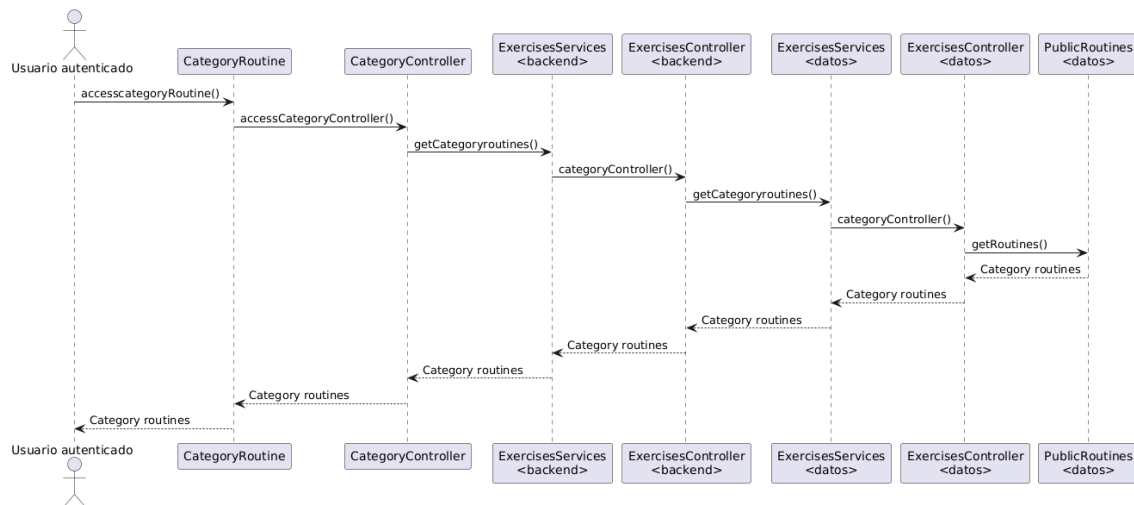


Figura 26: Realización de caso de uso [UC-012] Acceder a categoría

- **[UC-013] Eliminar rutina de categoría**

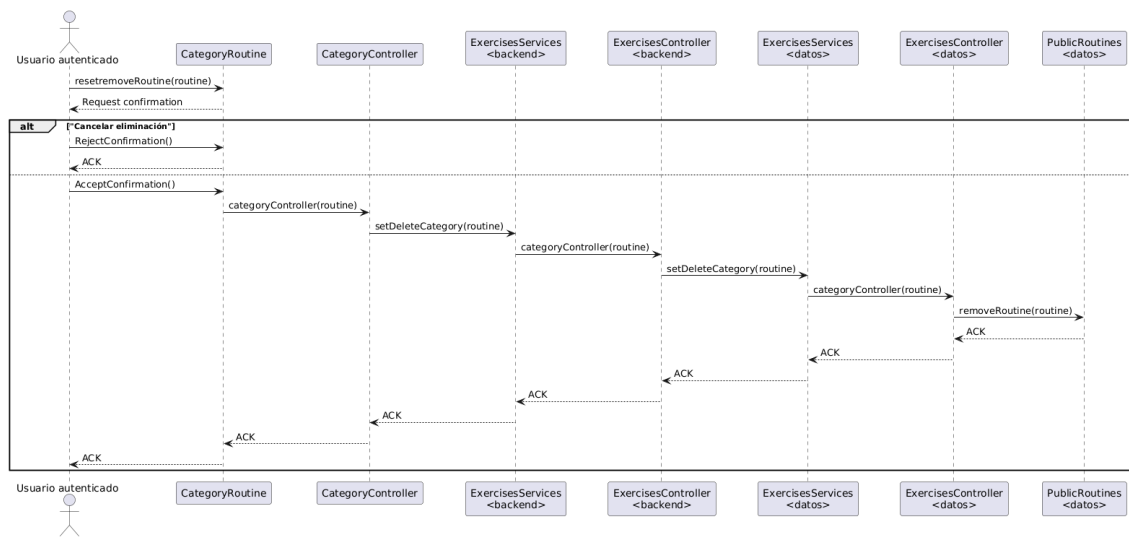


Figura 27: Realización de caso de uso [UC-013] Eliminar rutina de categoría

- **[UC-014] Limpiar categoría**

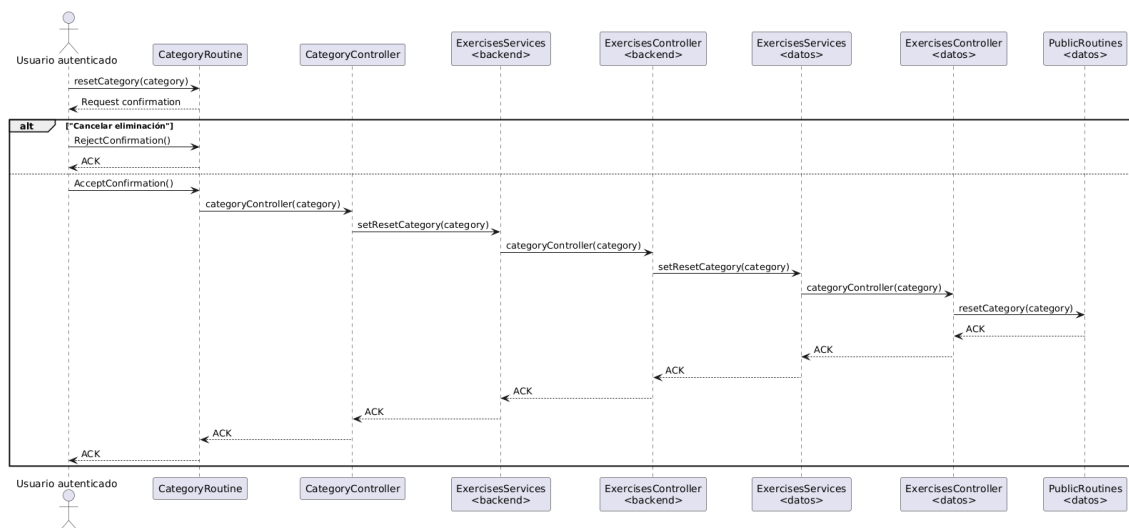


Figura 28: Realización de caso de uso [UC-014] Limpiar categoría

- **[UC-015] Mostrar historial de rutinas**

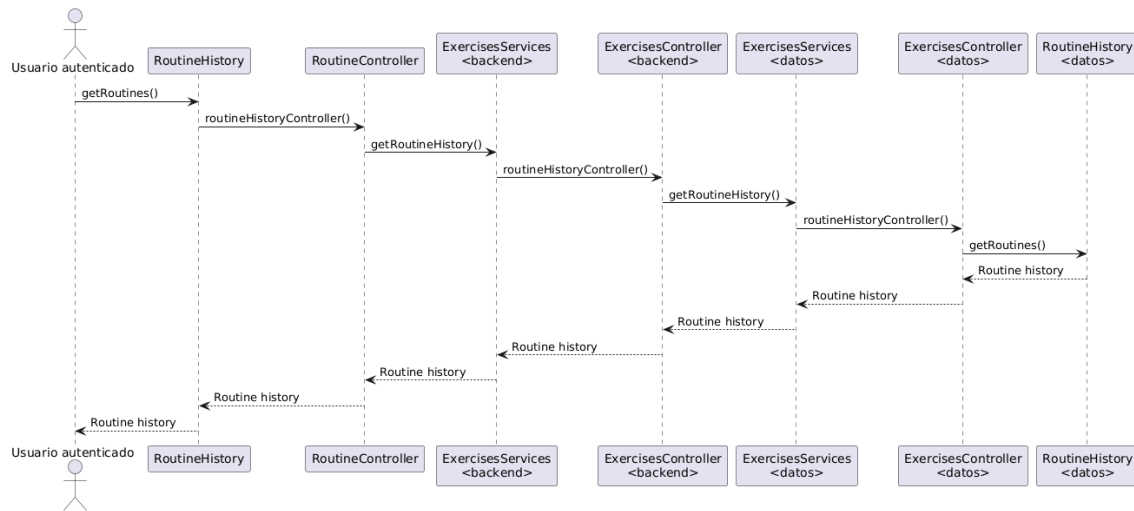


Figura 29: Realización de caso de uso [UC-015] Mostrar historial de rutinas

- **[UC-016] Limpiar historial de rutinas**

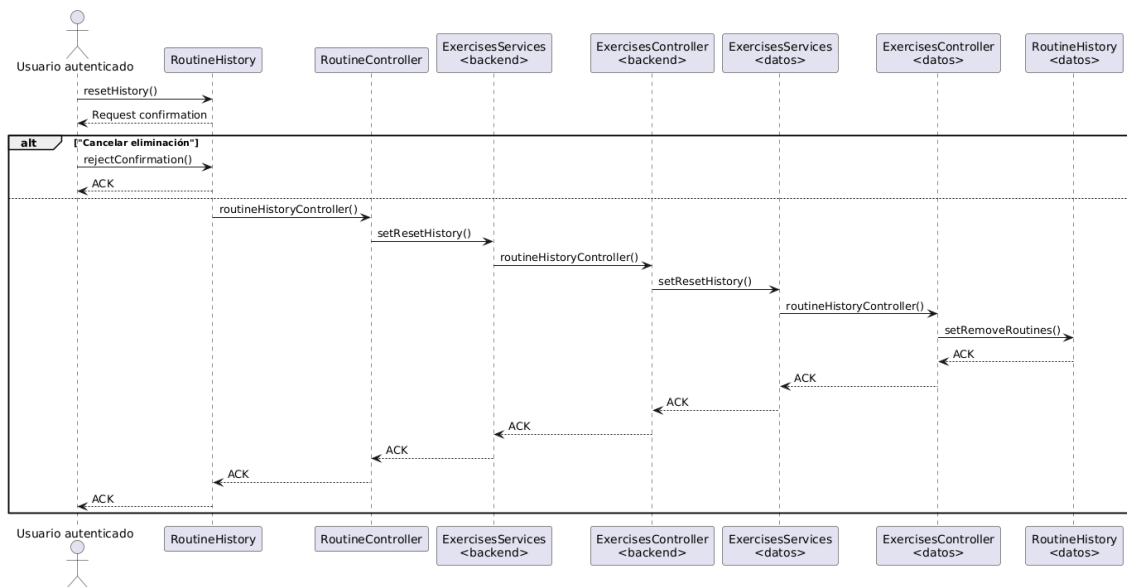


Figura 30: Realización de caso de uso [UC-016] Limpiar historial de rutinas

- [UC-017] Ejecutar una rutina

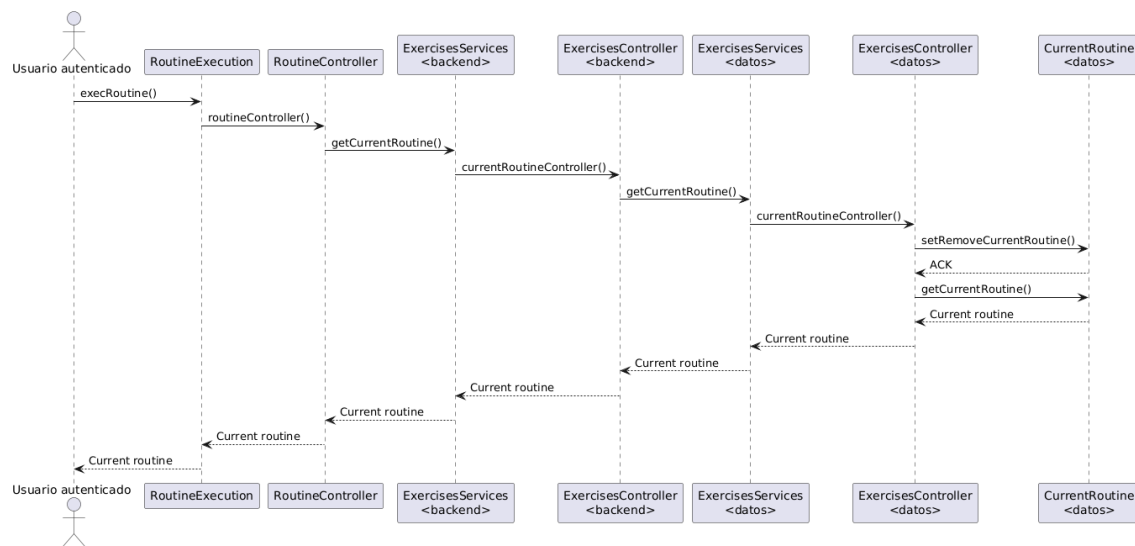


Figura 31: Realización de caso de uso [UC-017] Ejecutar una rutina

- [UC-018] Reconocer ejercicio

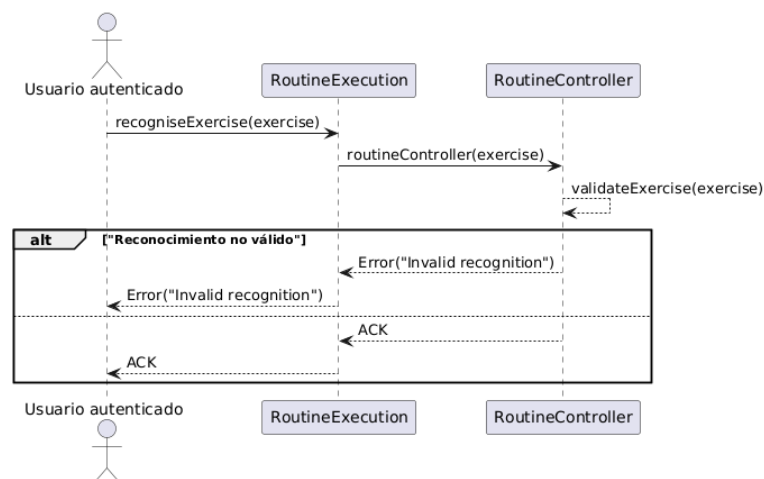


Figura 32: Realización de caso de uso [UC-018] Reconocer ejercicio

- [UC-019] Terminar rutina

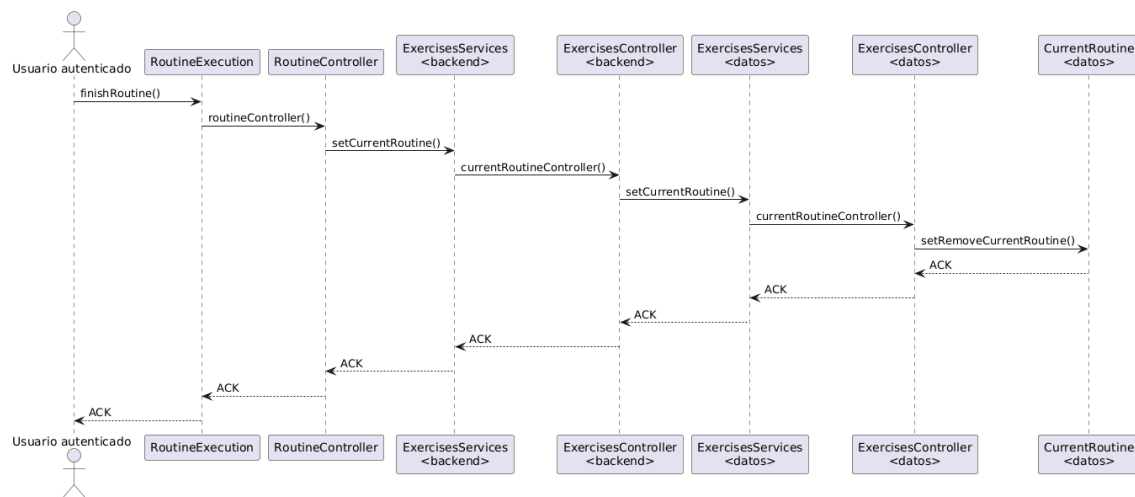


Figura 33: Realización de caso de uso [UC-019] Terminar rutina

5.4. GESTIÓN DE EJERCICIOS

- [UC-020] Mostrar ejercicios

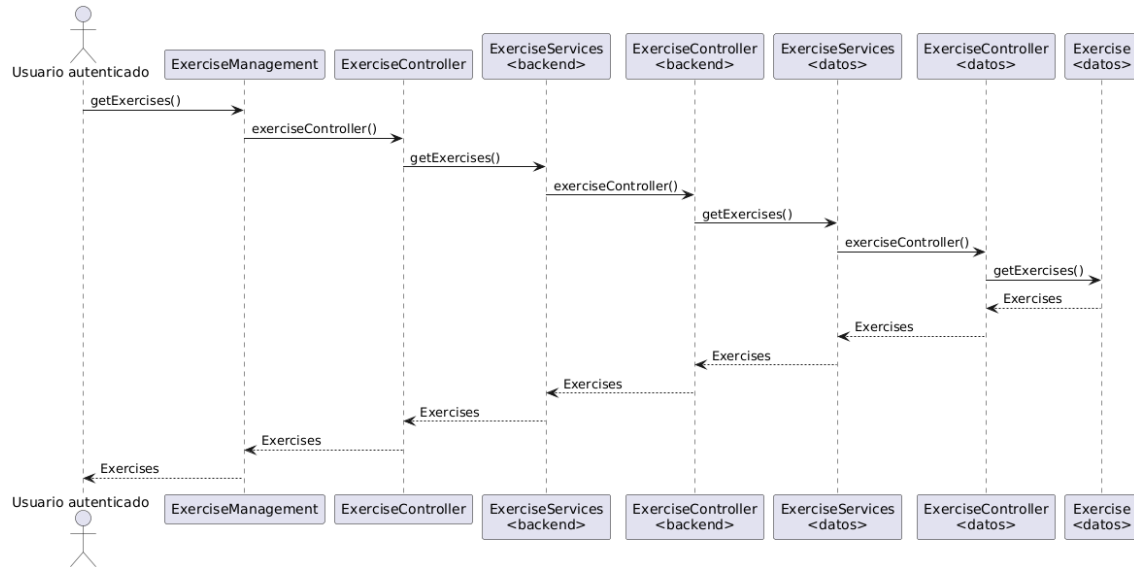


Figura 34: Realización de caso de uso [UC-020] Mostrar ejercicios

- [UC-021] Añadir ejercicio

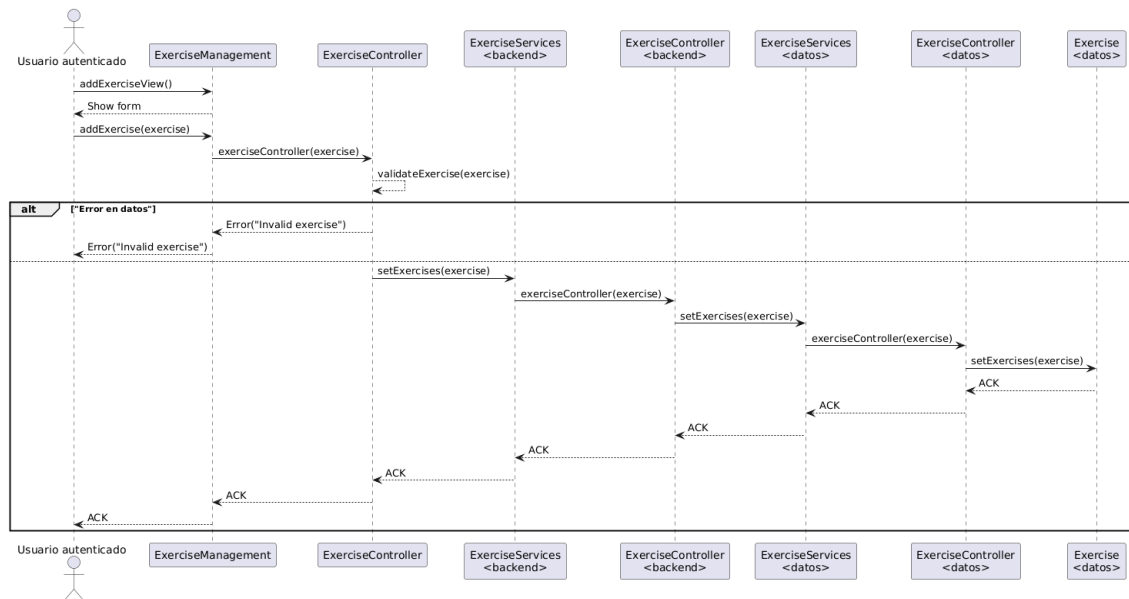


Figura 35: Realización de caso de uso [UC-021] Añadir ejercicio

- **[UC-022] Modificar ejercicio**

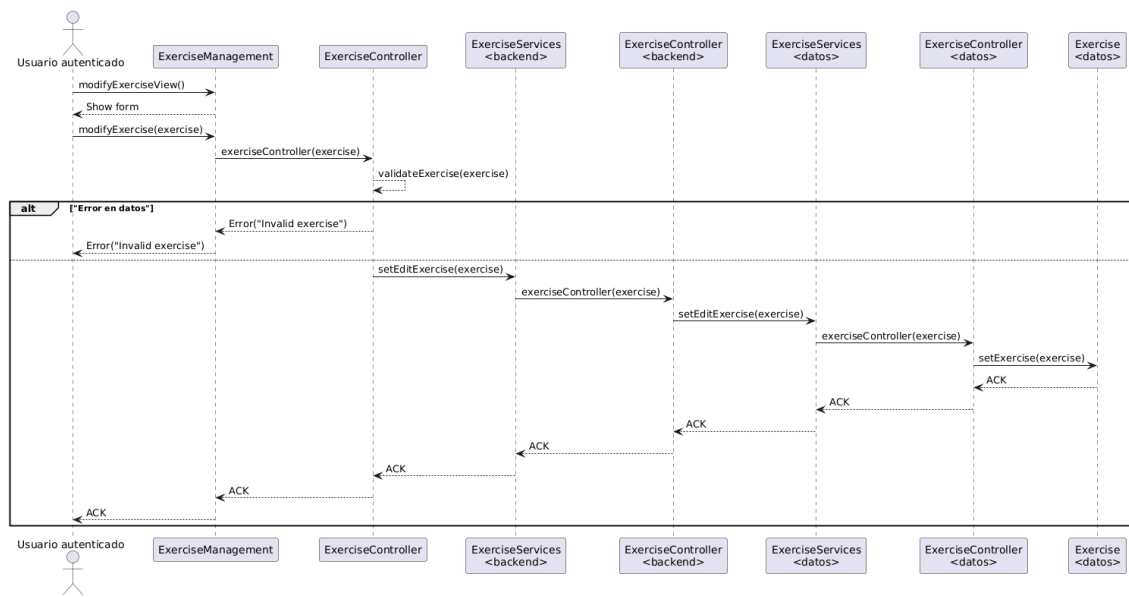


Figura 36: Realización de caso de uso [UC-022] Modificar ejercicio

- **[UC-023] Eliminar ejercicio**

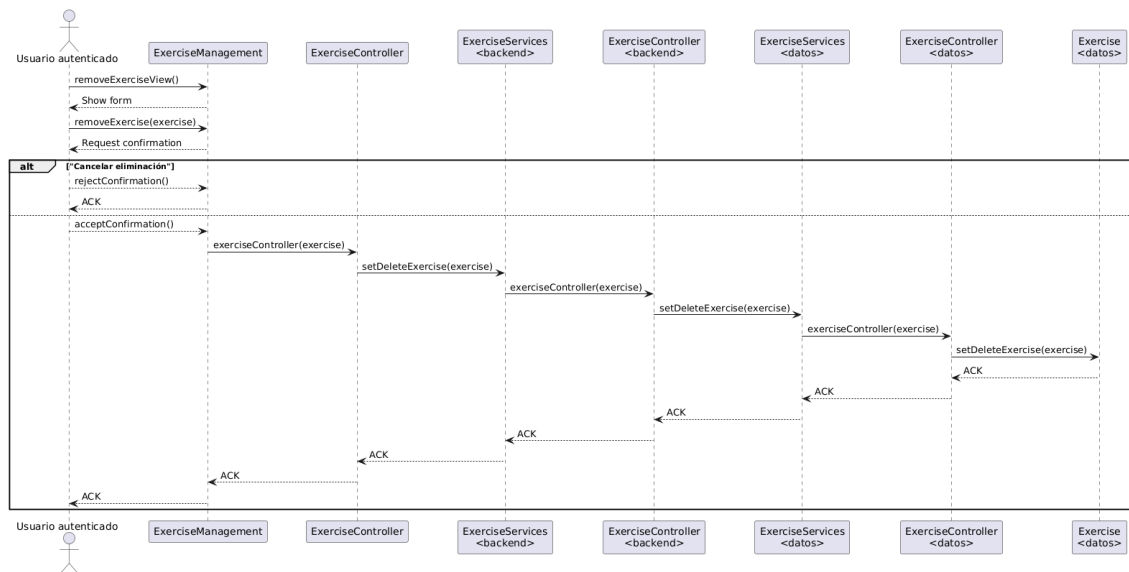


Figura 37: Realización de caso de uso [UC-023] Eliminar ejercicio

5.5. GESTIÓN DE ESTADÍSTICAS

- [UC-024] Mostrar estadísticas

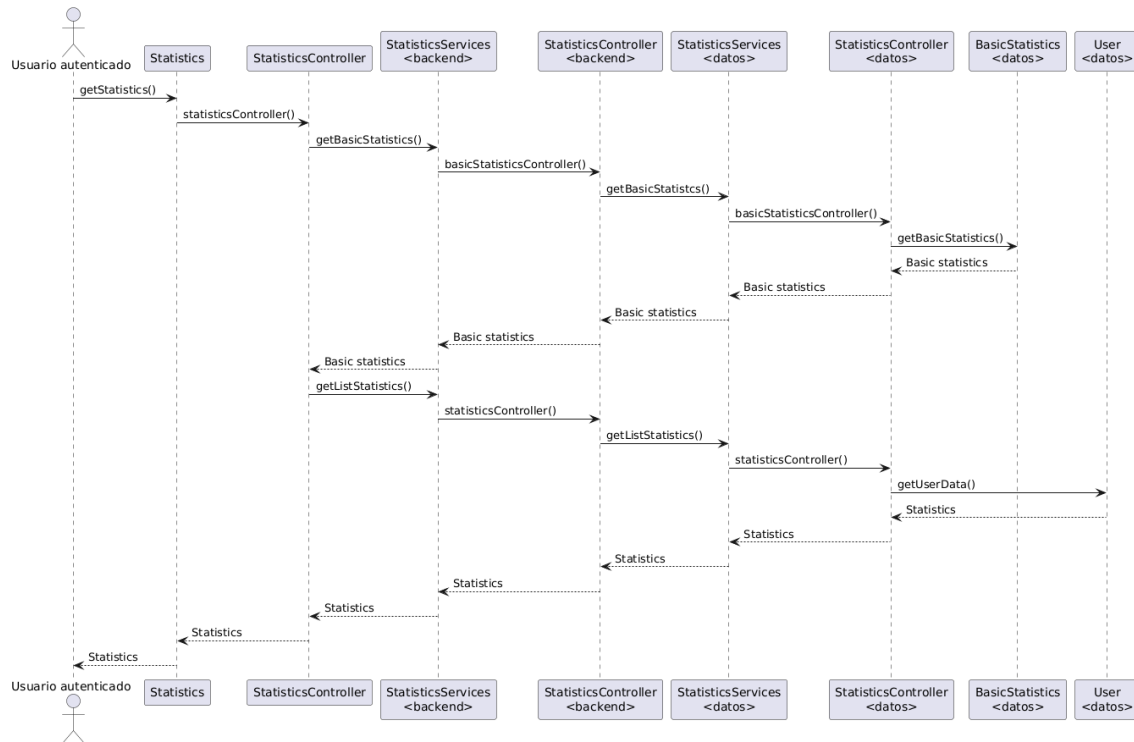


Figura 38: Realización de caso de uso [UC-024] Mostrar estadísticas

- [UC-025] Limpiar estadísticas

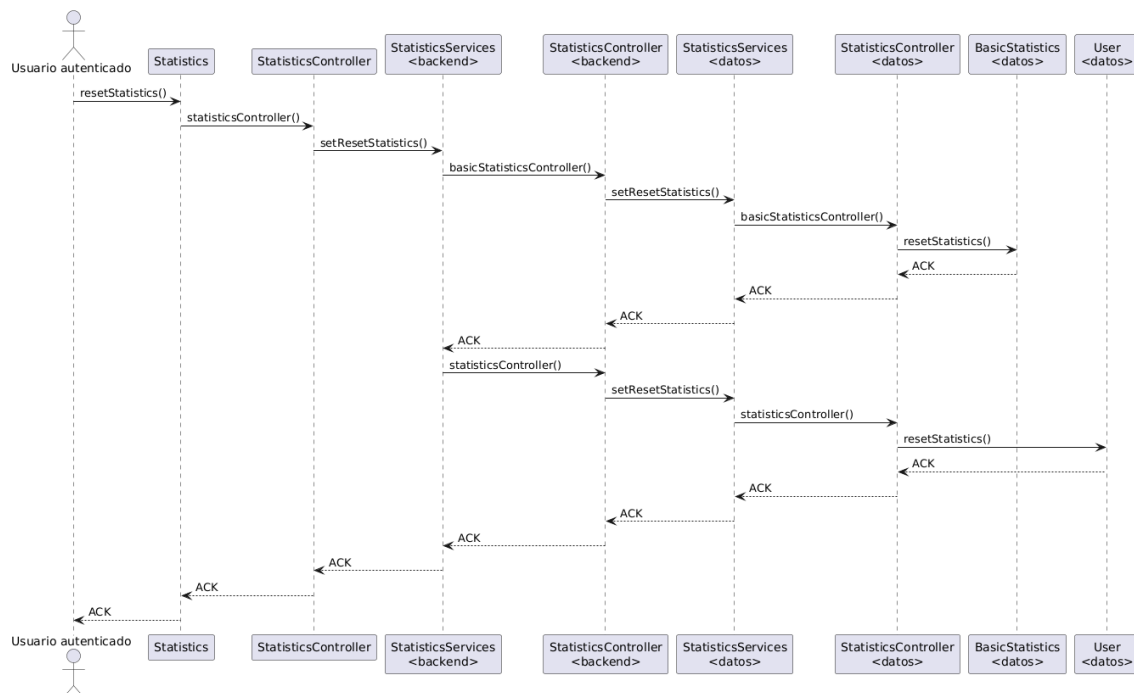


Figura 39: Realización de caso de uso [UC-025] Limpiar estadísticas

6. DISEÑO DE LA BASE DE DATOS

A continuación, se presenta la estructura organizativa de la base de datos del sistema. Se detallarán todas las tablas y atributos junto con sus respectivos tipos de datos. La base de datos utilizada será de tipo SQL, concretamente MariaDB.

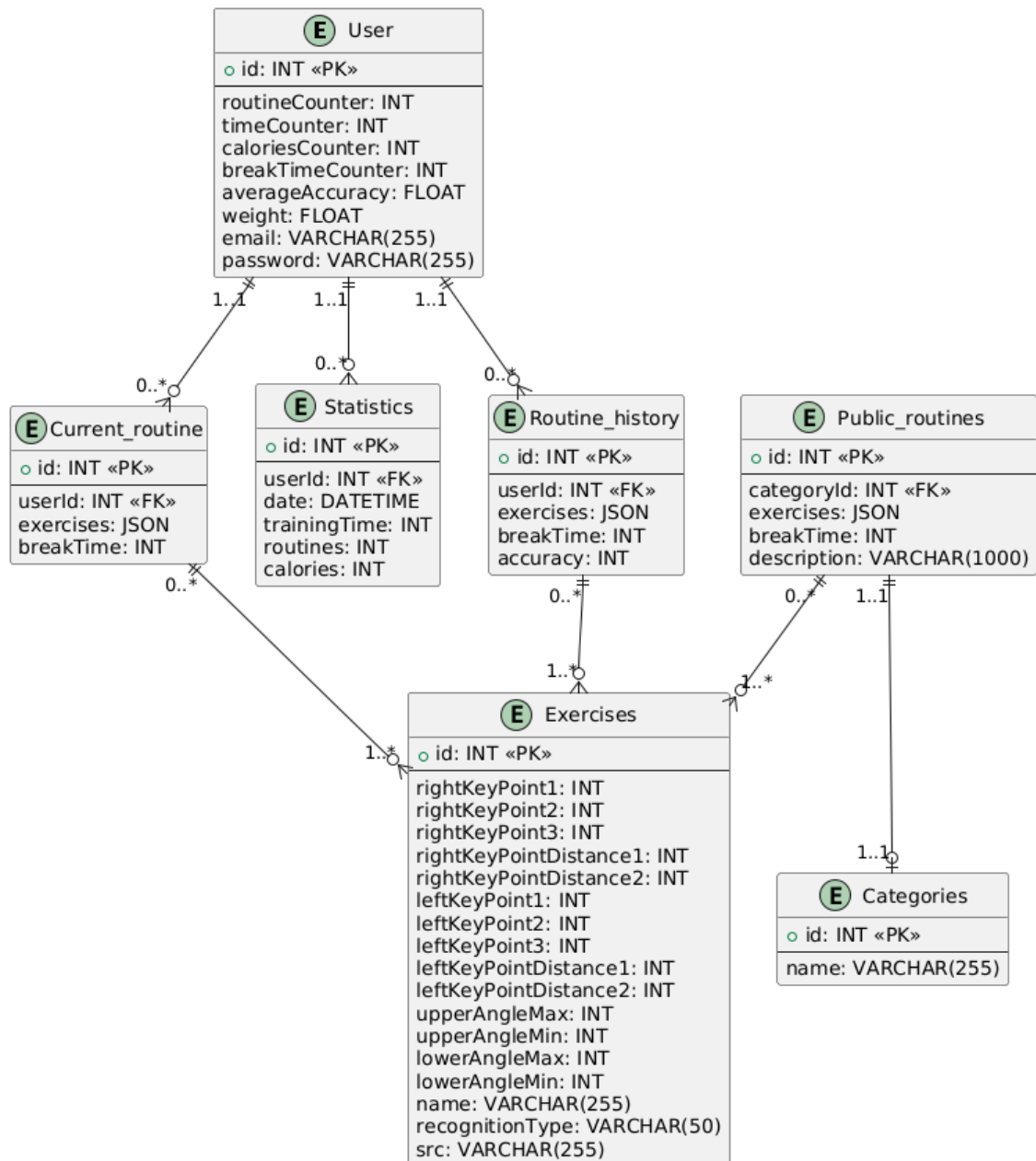


Figura 40: Diseño de la base de datos

7. MODELO DE DESPLIEGUE

El modelo de despliegue representa la distribución física del sistema, especificando cómo se asigna la funcionalidad a los distintos nodos de procesamiento. La Figura 41 ilustra la implementación de este diagrama.

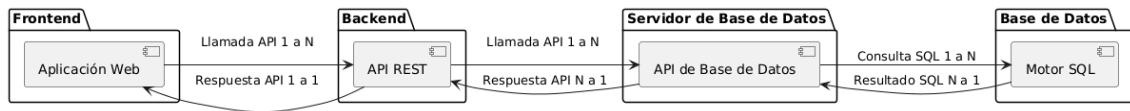


Figura 41: Diagrama de despliegue

El modelo de despliegue describe la estructura de un sistema durante su ejecución y está compuesto por cuatro nodos:

- **Frontend:** Parte del sistema que interactúa directamente con el usuario, formando la interfaz gráfica y los elementos que el usuario puede ver y manipular.
- **Backend:** Parte del sistema que se encarga de la lógica, procesamiento de datos y gestión del servidor. Es la capa que funciona detrás de escena, permitiendo que el frontend opere correctamente.
- **Servidor de base de datos:** Sistema que proporciona servicios de almacenamiento, gestión y acceso a bases de datos. Su función es responder las solicitudes del backend para operar en la base de datos garantizando integridad, seguridad y disponibilidad.
- **Base de datos:** Conjunto organizado de datos estructurados que se almacenan y gestionan de manera eficiente para facilitar su acceso, manipulación y actualización.

8. REFERENCIAS

- [1] Sommerville, I. (2005). *Ingeniería del software*. Pearson educación.