

# APLICACIÓN PARA LA COMUNICACIÓN ENTRE CENTROS EDUCATIVOS Y FAMILIAS

## Anexo III: Análisis de Requisitos



VNiVERSiDAD  
D SALAMANCA

Septiembre de 2023

**Trabajo de fin de Grado**

**Tutor:**

María Navarro Cáceres

**Alumno:**

Raúl Rodríguez Villanueva

## Tabla de Contenido

1. INTRODUCCIÓN.....	4
2. MODELO DEL DOMINIO .....	4
3. REALIZACION DE CASOS DE USO EN EL MODELO DE ANÁLISIS.....	5
4. PROPUESTA DE ARQUITECTURA.....	20
5. REFERENCIAS.....	22

## Imágenes

Imagen 1. Diagrama de clases .....	4
Imagen 2. UC-001 Registrarse .....	5
Imagen 3. UC-002 Iniciar Sesión.....	6
Imagen 4. UC-003 Recuperar contraseña.....	6
Imagen 5. UC-004 Cerrar sesión.....	7
Imagen 6. UC-005 Ver lista de exámenes .....	7
Imagen 7. UC-006 Añadir examen.....	8
Imagen 8. UC-007 Modificar examen.....	9
Imagen 9. UC-008 Eliminar examen .....	10
Imagen 10. UC-009 Ver calendario de actividades.....	10
Imagen 11. UC-010 Avanzar semana en el calendario .....	11
Imagen 12. UC-011 Ir semana atrás en el calendario.....	11
Imagen 13. UC-012 Añadir actividad.....	12
Imagen 14. UC-013 Modificar actividad.....	12
Imagen 15. UC-014 Eliminar actividad .....	13
Imagen 16. UC-015 Ver resultados.....	13
Imagen 17. UC-016 Seleccionar jornada .....	14
Imagen 18. UC-017 Añadir jornada .....	14
Imagen 19. UC-018 Eliminar jornada .....	15
Imagen 20. UC-019 Añadir resultado .....	16
Imagen 21. UC-020 Modificar resultado .....	16
Imagen 22. UC-021 Eliminar resultado.....	17
Imagen 23. UC-022 Ver información de contacto .....	17
Imagen 24. UC-023 Ver ajustes .....	18
Imagen 25. UC-024 Modificar datos del usuario.....	18
Imagen 26. UC-025 Modificar contraseña.....	19
Imagen 27. UC-026 Asignar permisos a usuario .....	20
Imagen 28. Propuesta de arquitectura.....	21

# 1. INTRODUCCIÓN

En este anexo se va a documentar el análisis de los requisitos redactados en el anterior anexo. Dentro de este documento, vamos a tener la siguiente estructura:

- Modelo del dominio: Representación de las clases que componen nuestro sistema y las relaciones que existen entre ellas. Para esto se utilizará un diagrama de clases.
- Realización de casos de uso en el modelo de análisis: Describe a través de diagramas la ejecución de los diferentes casos de uso redactados. Para ello se utilizarán diagramas de secuencia.
- Propuesta de arquitectura: Representación de la arquitectura seguida en el proyecto. Para esto se utilizará un diagrama de paquetes.

# 2. MODELO DEL DOMINIO

En este apartado se pretende representar a través de un diagrama de clases las principales clases del proyecto junto con alguno de sus atributos. También se presentan las relaciones entre estas clases.

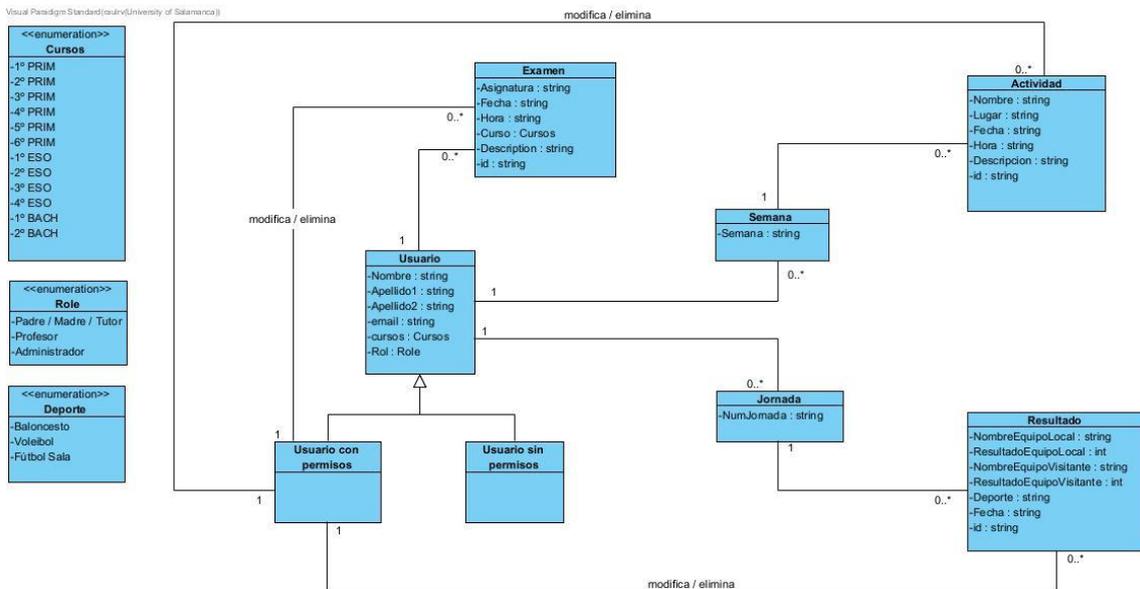


Imagen 1. Diagrama de clases

En la parte izquierda del diagrama se encuentran las enumeraciones, las cuales se van a utilizar, sobre todo, en los Spinners de Android Studio, siendo estos unos desplegables con diferentes opciones las cuales pueden ser seleccionadas para establecer opciones predeterminadas.

Tenemos una clase principal “Usuario”, de la cual hay una relación de dependencia con dos tipos de usuarios, el usuario con permisos (dentro de los cuales se encuentran aquellos capaces de modificar y eliminar los diferentes elementos de la aplicación. Aquí estarán los profesores y administradores) y el usuario sin permisos (aquí estarán los usuarios registrados pero que solamente tienen la capacidad de visualización, sin poder modificar nada).

Un usuario puede ver 0 o más jornadas, en caso de estar hablando de resultados, por ello mantiene una relación 1 – 0..\*, al igual que ocurre con las semanas dentro de las actividades. A su vez cada jornada puede tener 0 o más resultados, dependiendo del caso.

### 3. REALIZACION DE CASOS DE USO EN EL MODELO DE ANÁLISIS

En este apartado se van a representar los diferentes casos de uso mediante diagramas de secuencia. En ellos, además de los actores que interactúan con el sistema, se van a ver diferentes tipos de clases:

- Clases de interfaz: Esta es la clase que está en contacto con el usuario, la que se encarga de la comunicación con él.
- Clases de control: Se encargan de realizar las funcionalidades del sistema, a partir de lo que el usuario le ha solicitado a la interfaz.
- Clases de entidad: Son las clases que se almacenan en el sistema, conteniendo la información.

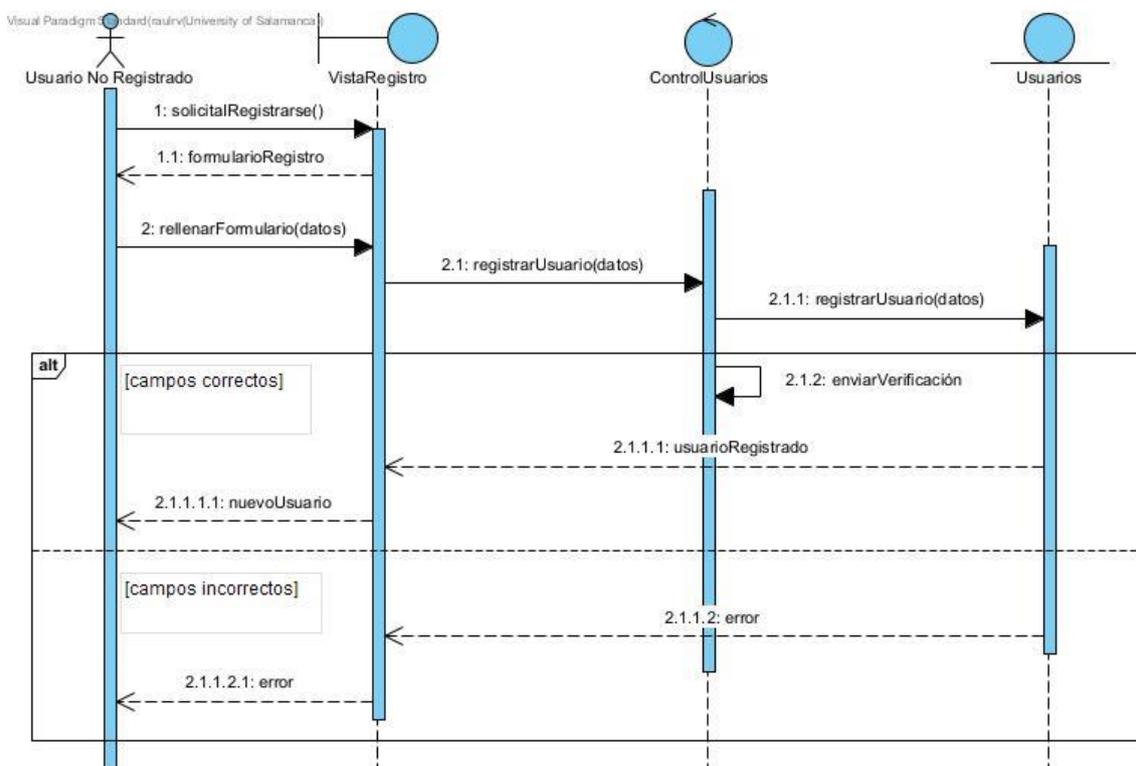


Imagen 2. UC-001 Registrarse

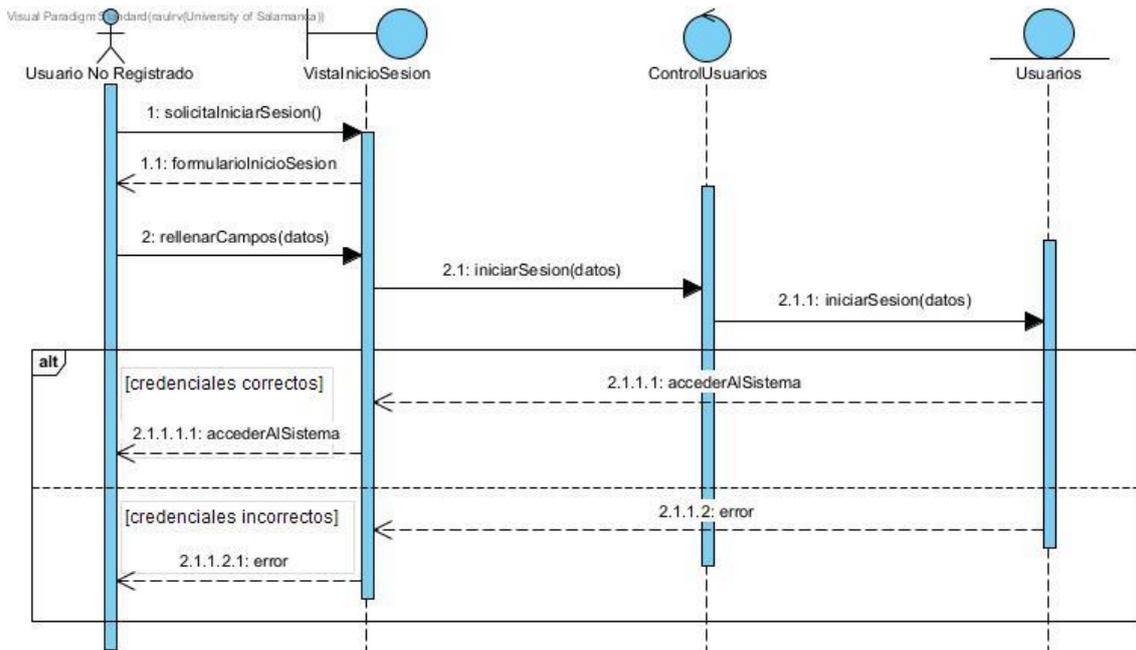


Imagen 3. UC-002 Iniciar Sesión

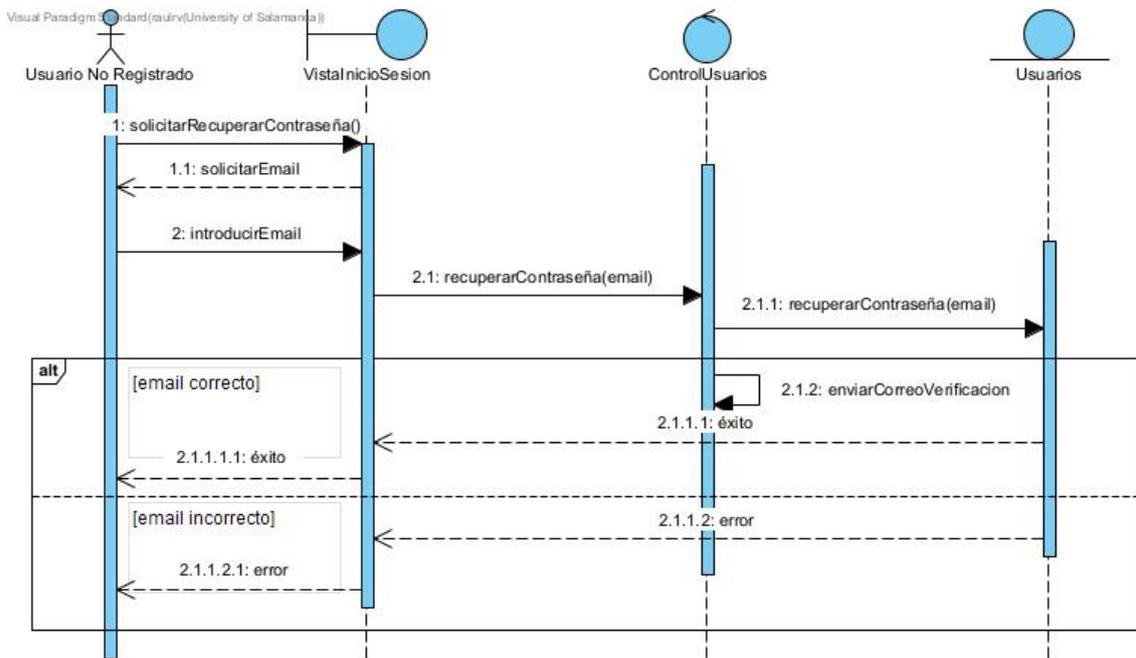


Imagen 4. UC-003 Recuperar contraseña

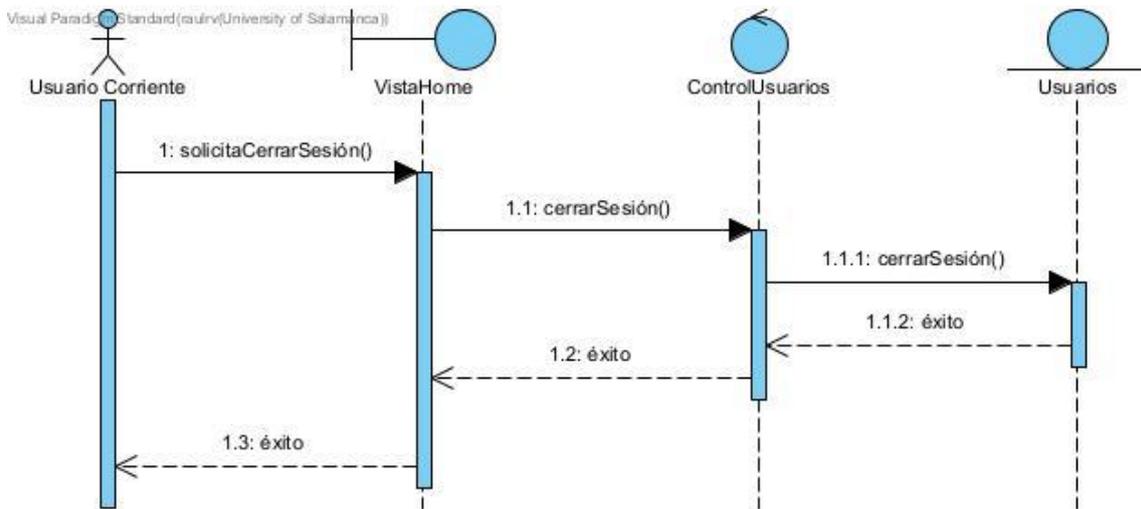


Imagen 5. UC-004 Cerrar sesión

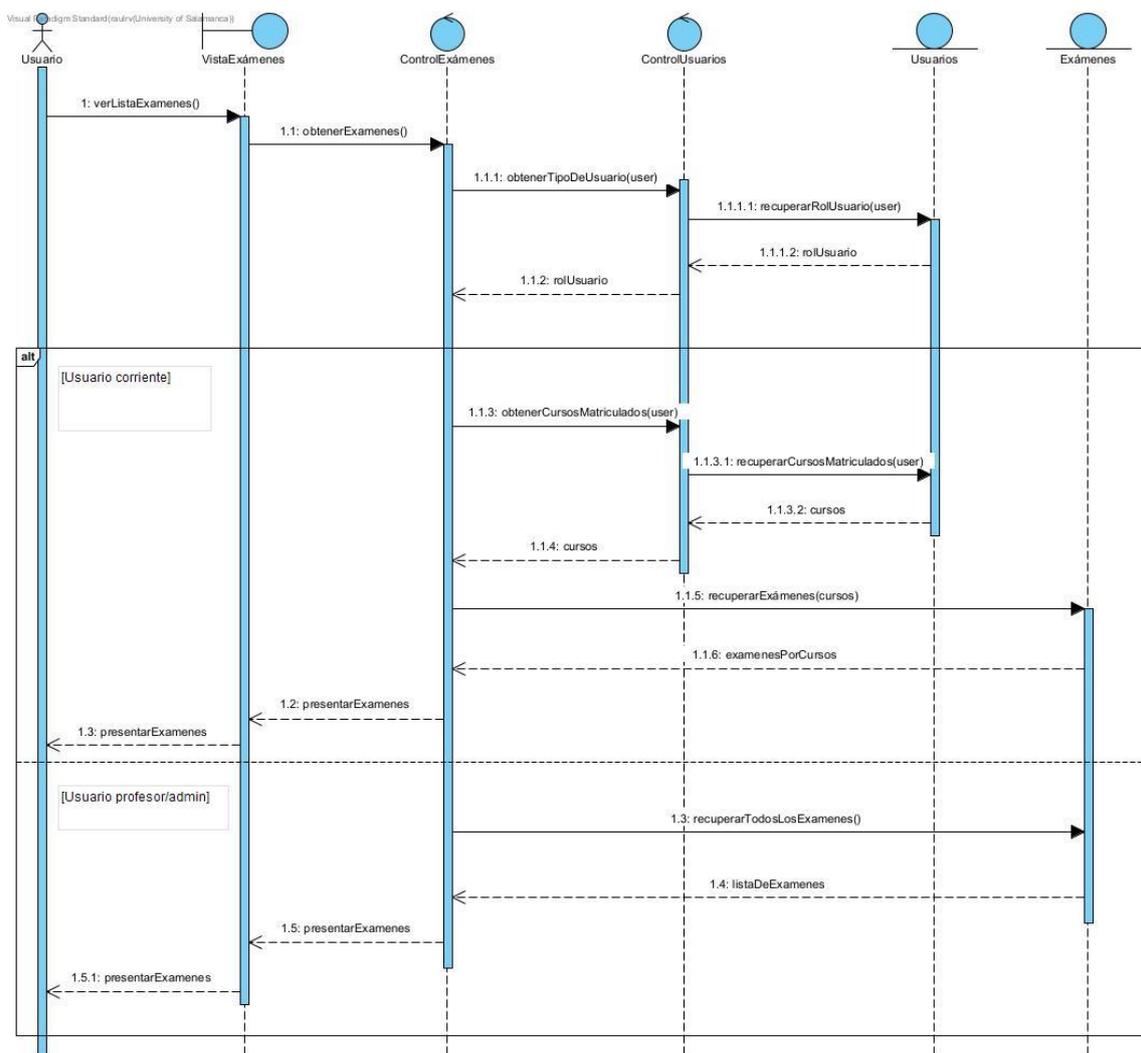


Imagen 6. UC-005 Ver lista de exámenes

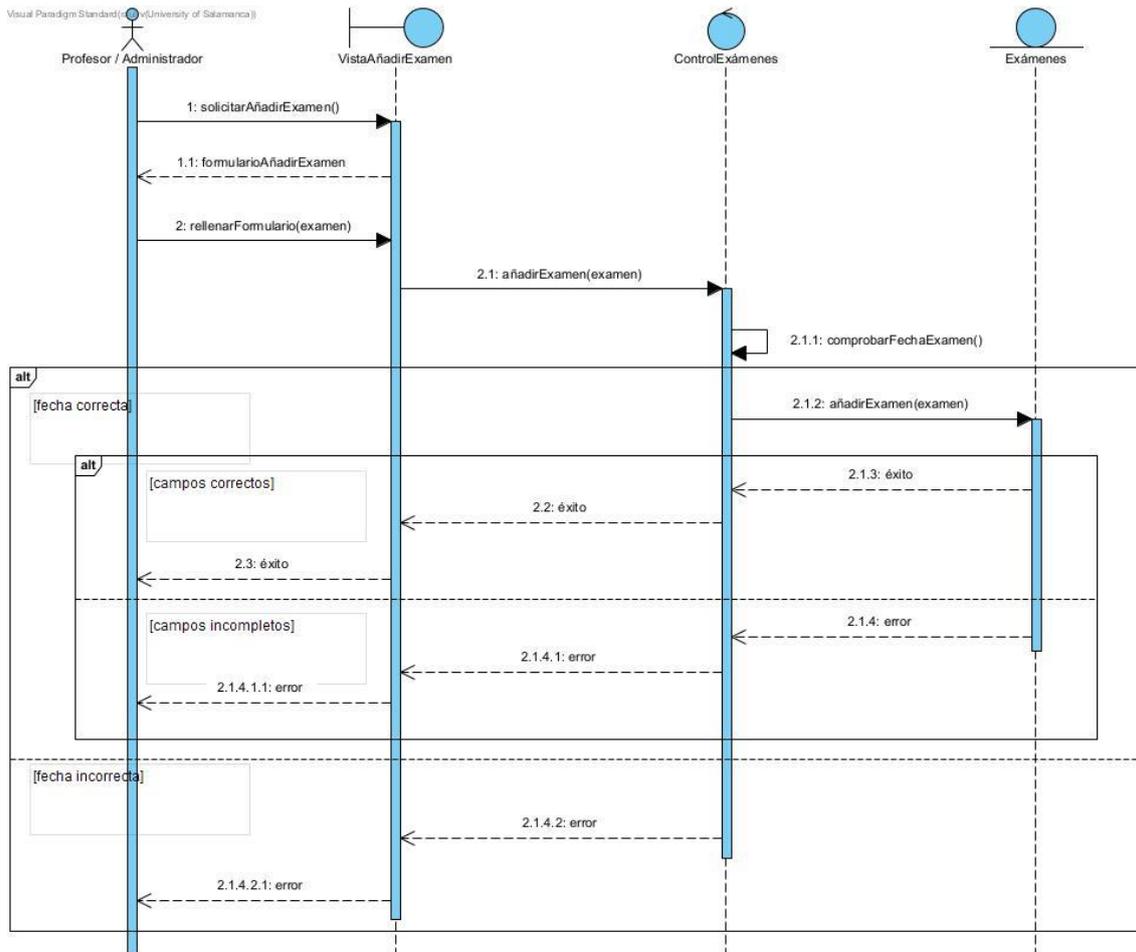


Imagen 7. UC-006 Añadir examen

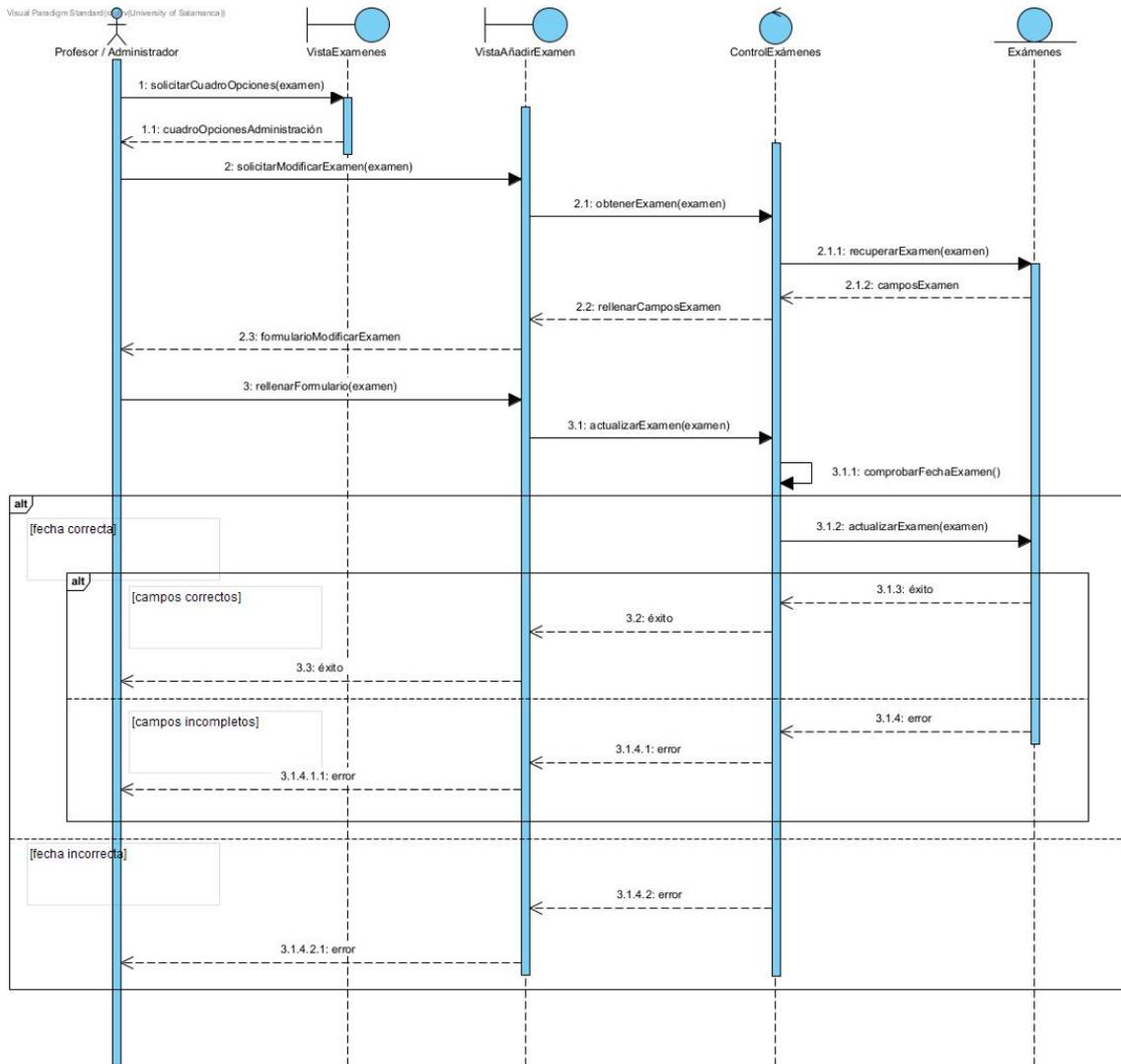


Imagen 8. UC-007 Modificar examen

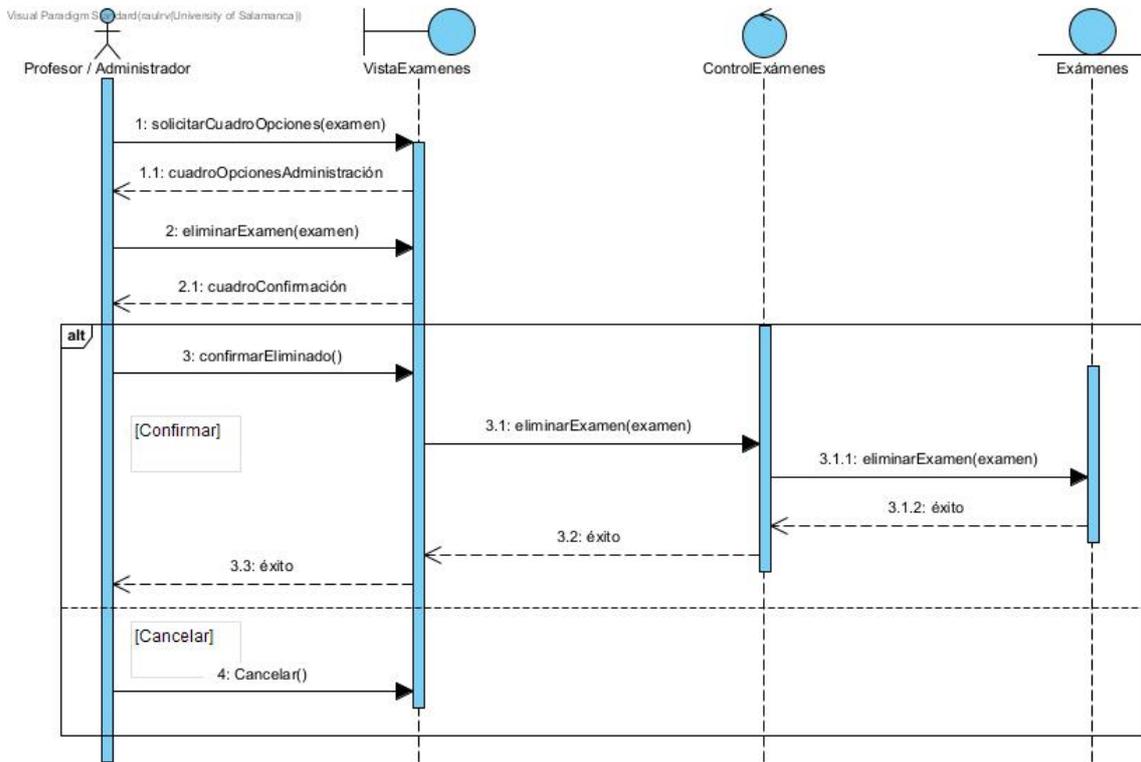


Imagen 9. UC-008 Eliminar examen

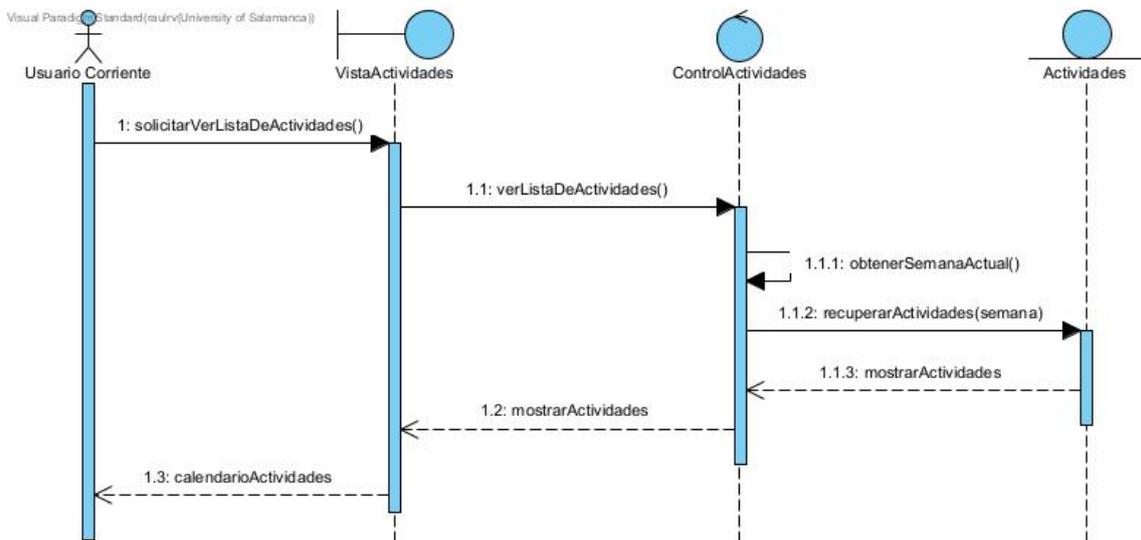


Imagen 10. UC-009 Ver calendario de actividades

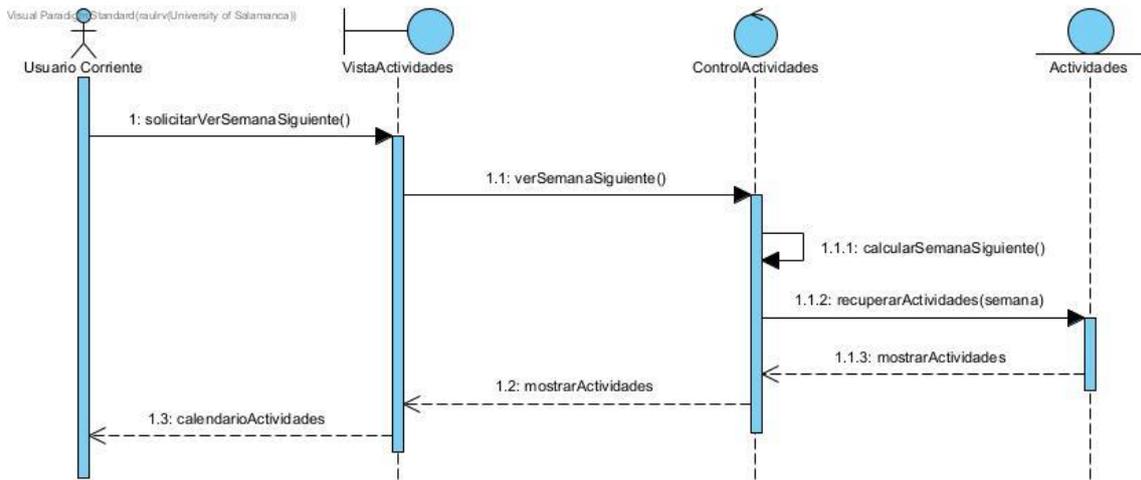


Imagen 11. UC-010 Avanzar semana en el calendario

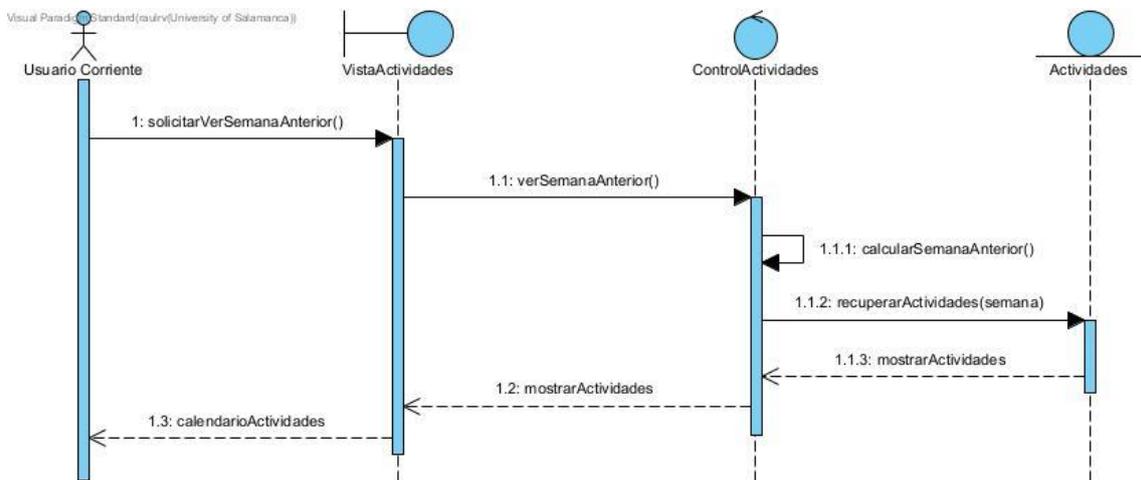


Imagen 12. UC-011 Ir semana atrás en el calendario

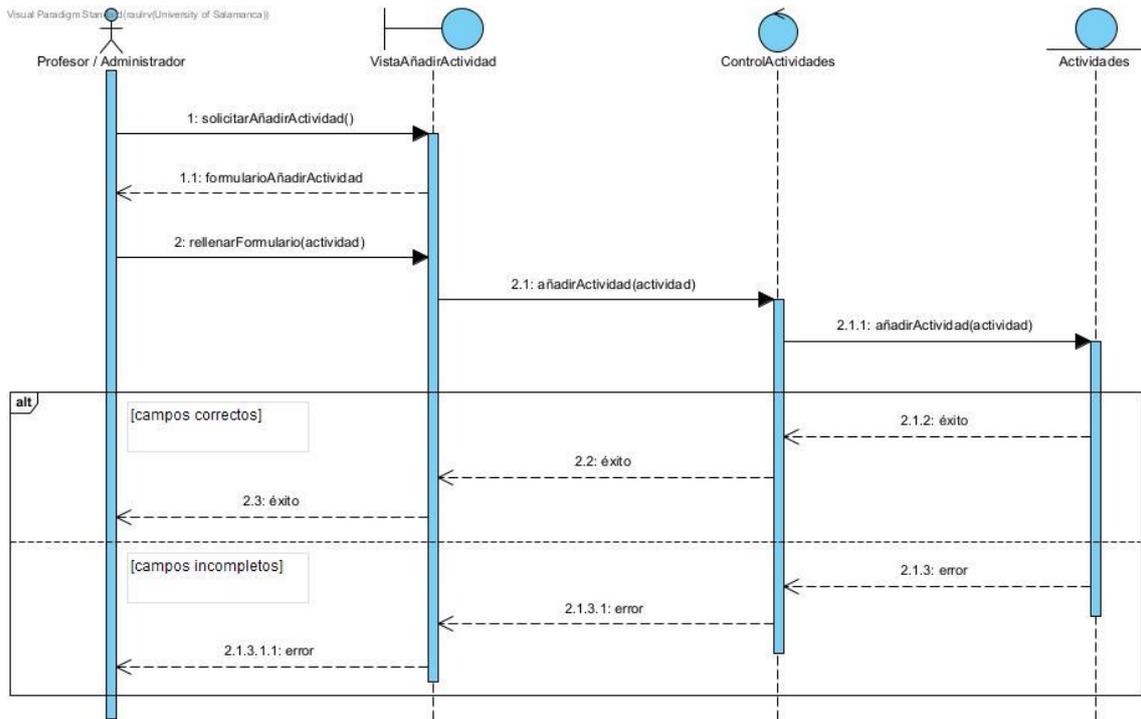


Imagen 13. UC-012 Añadir actividad

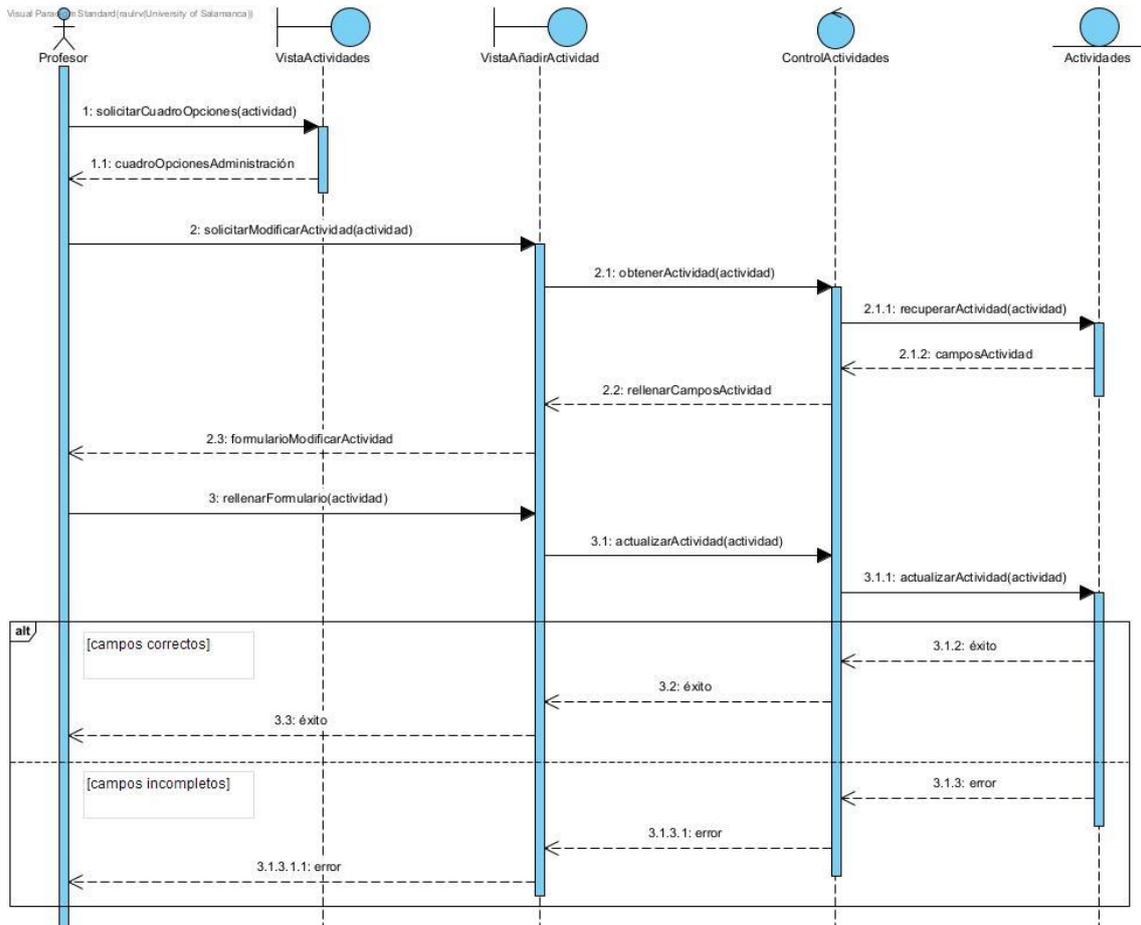


Imagen 14. UC-013 Modificar actividad

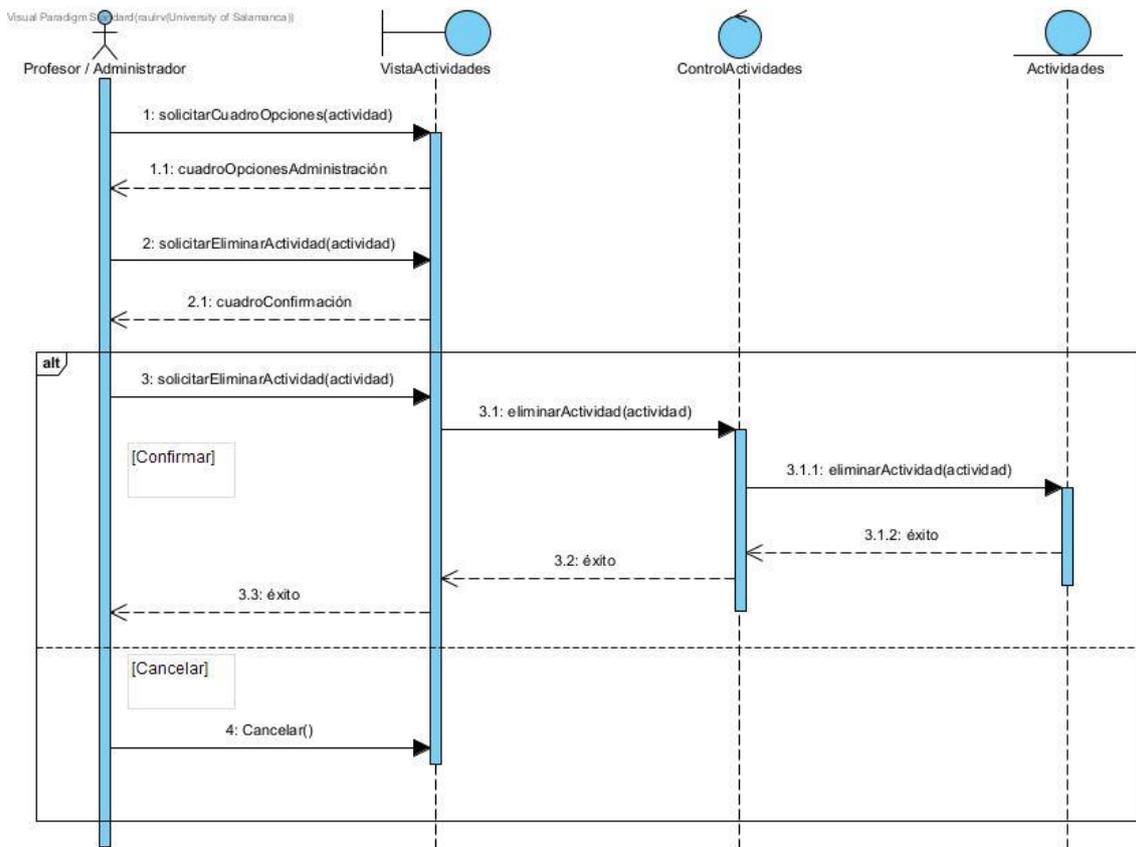


Imagen 15. UC-014 Eliminar actividad

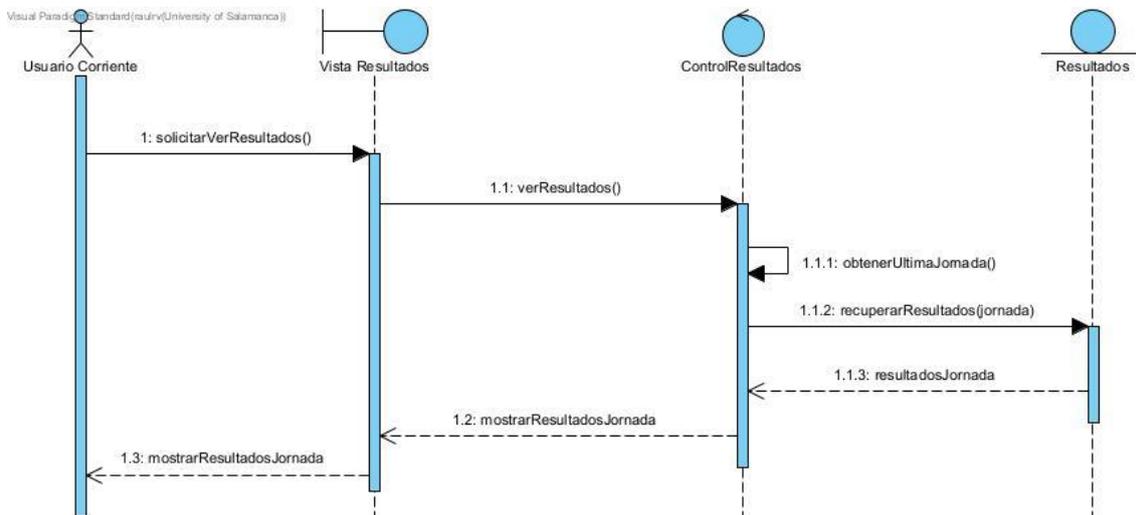


Imagen 16. UC-015 Ver resultados

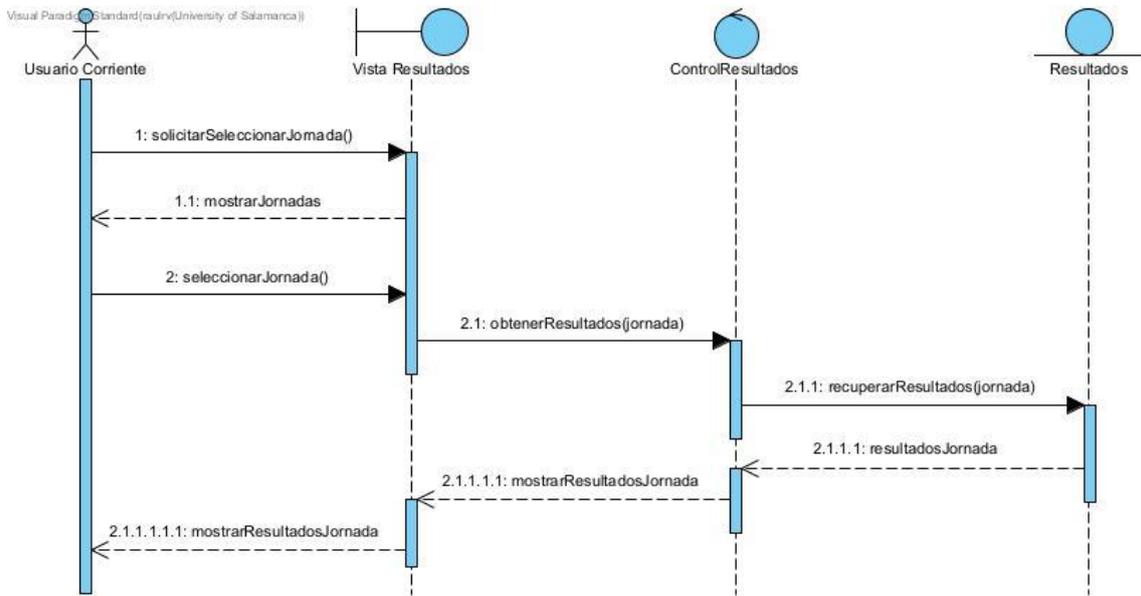


Imagen 17. UC-016 Seleccionar jornada

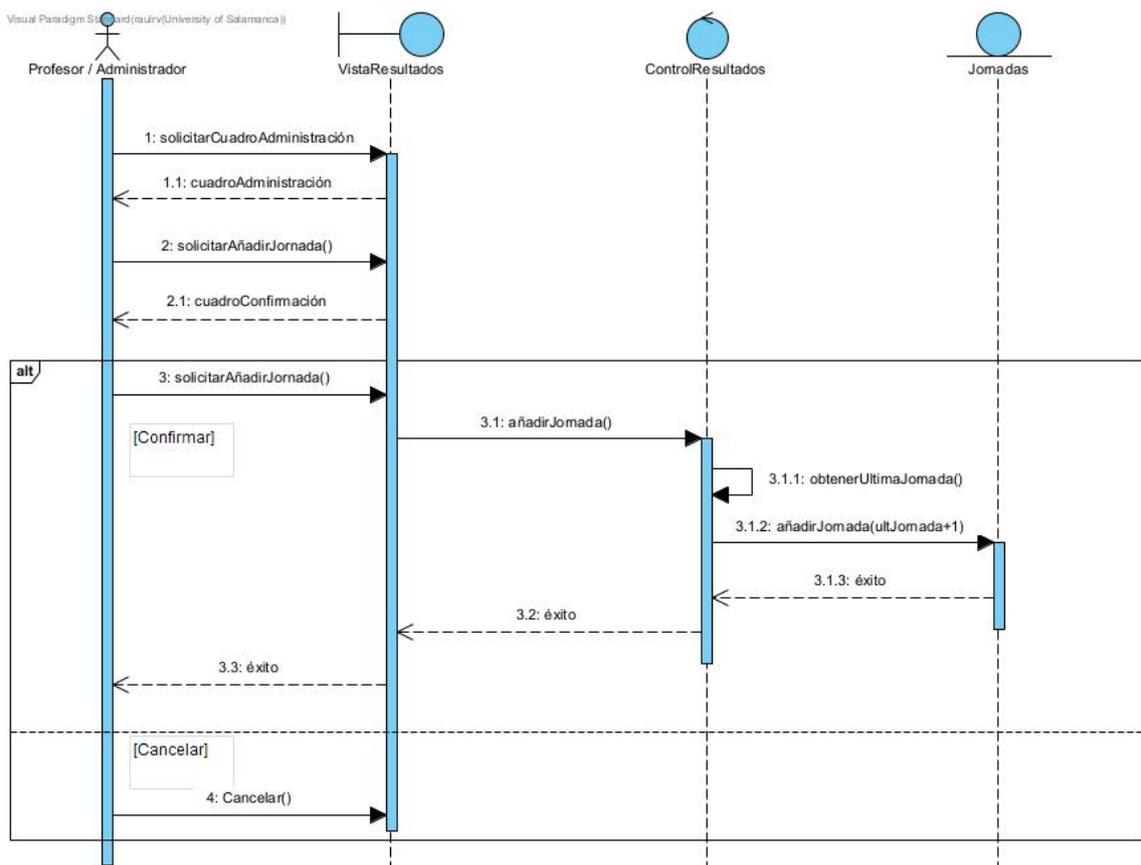


Imagen 18. UC-017 Añadir jornada

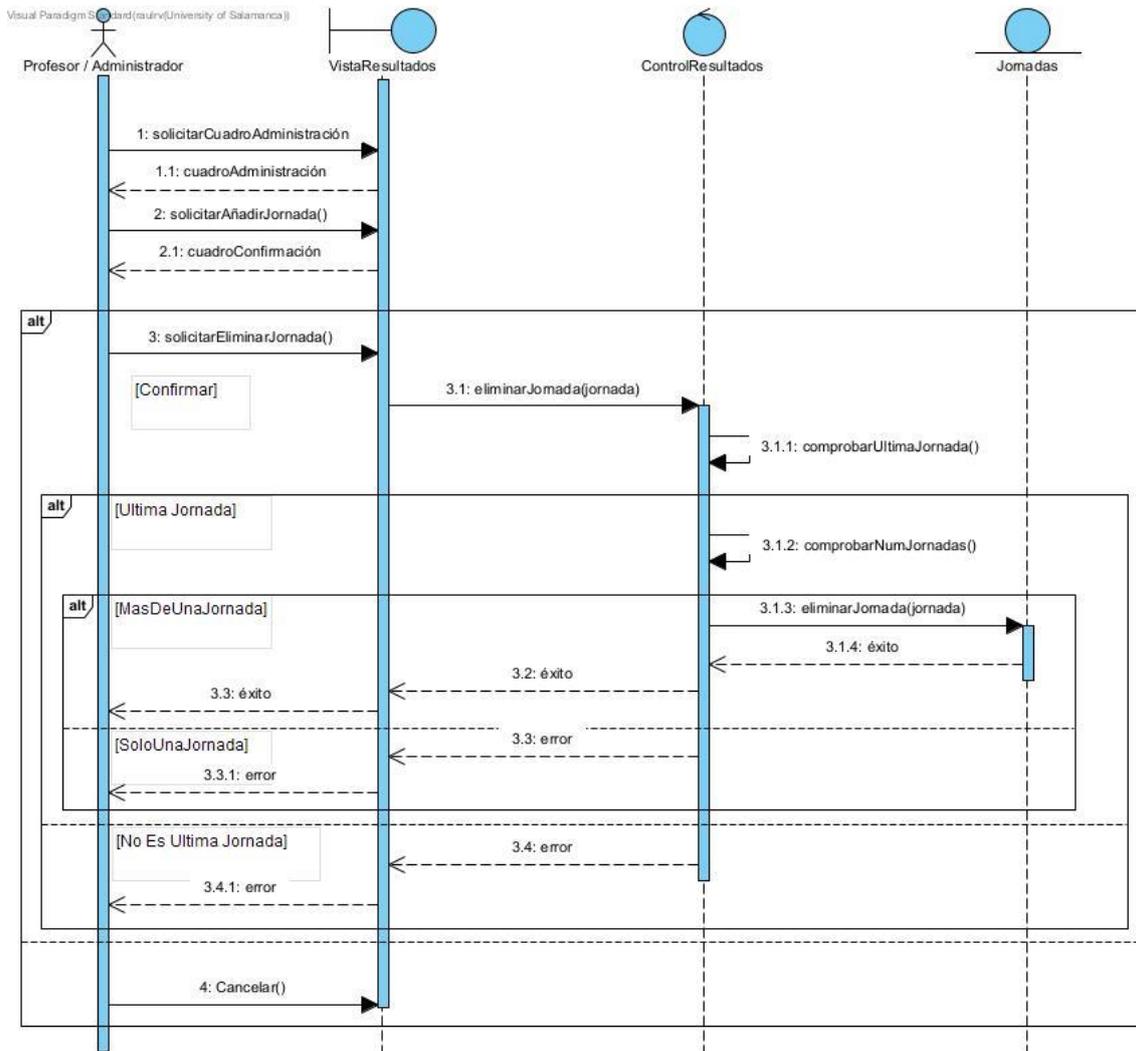


Imagen 19. UC-018 Eliminar jornada

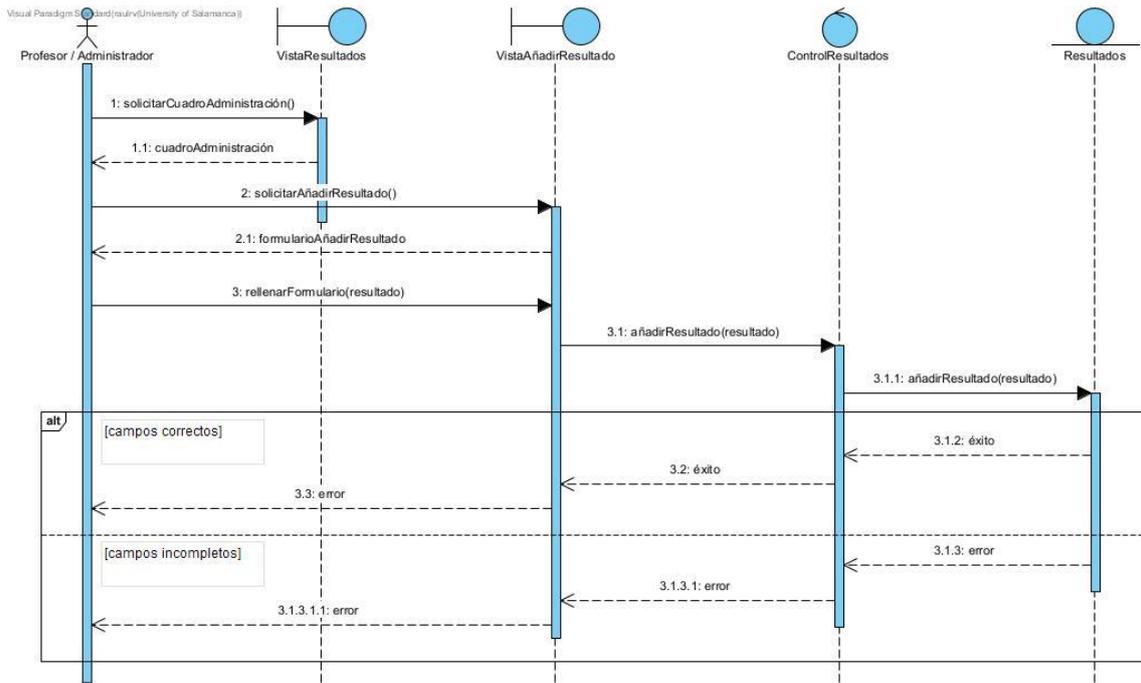


Imagen 20. UC-019 Añadir resultado

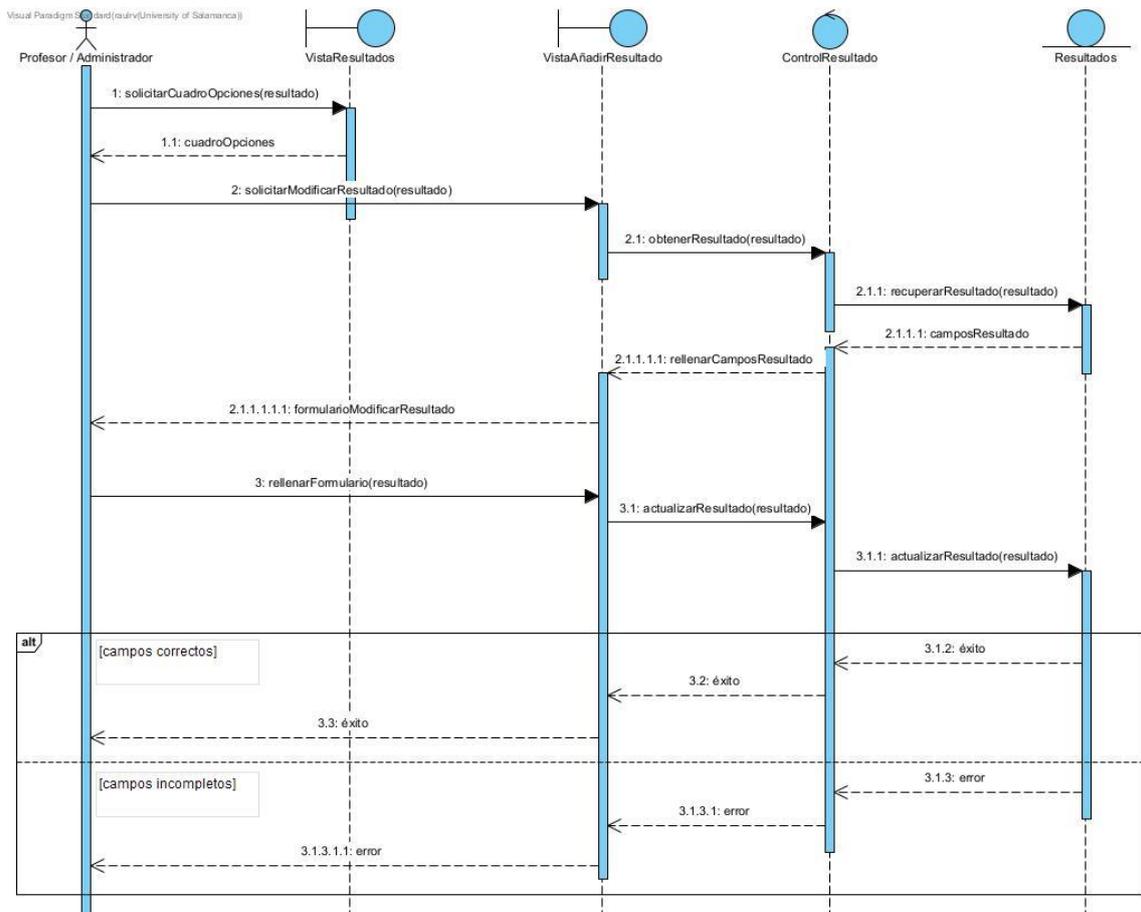


Imagen 21. UC-020 Modificar resultado

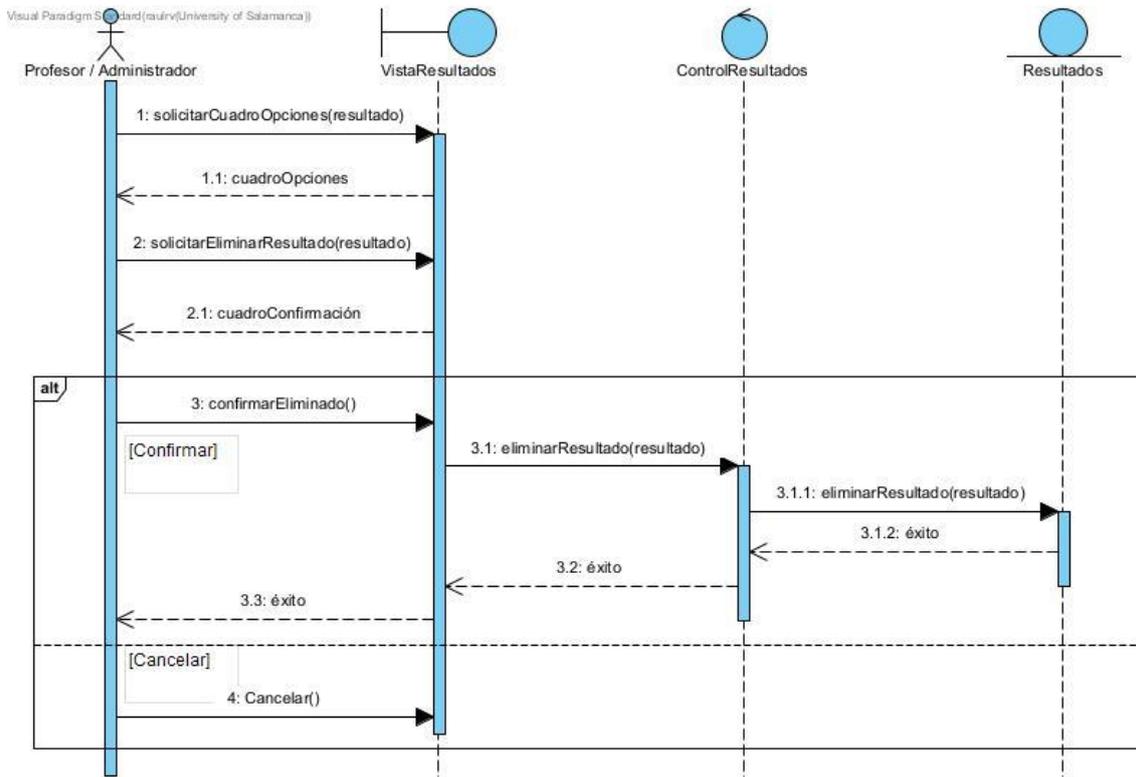


Imagen 22. UC-021 Eliminar resultado

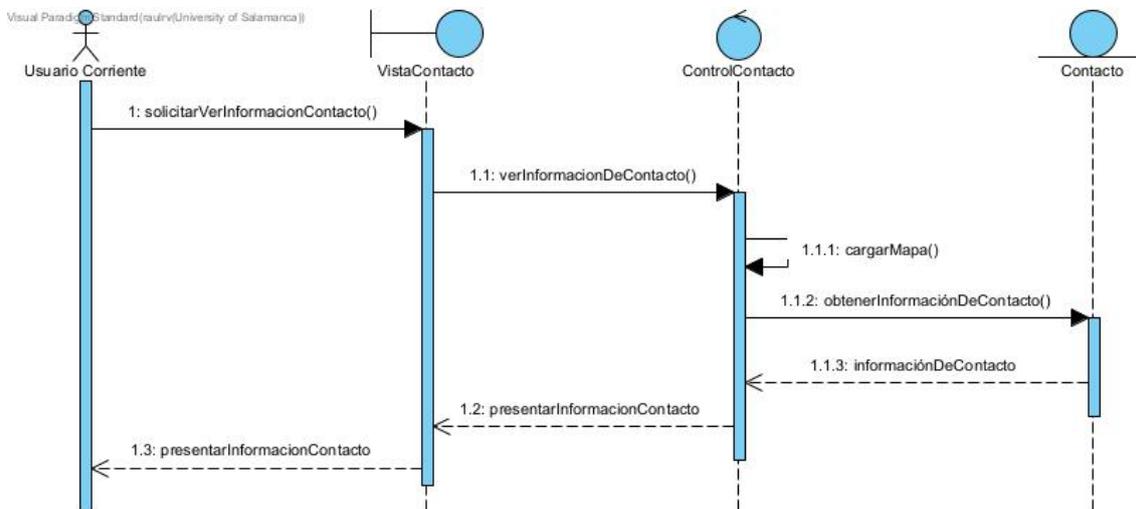


Imagen 23. UC-022 Ver información de contacto

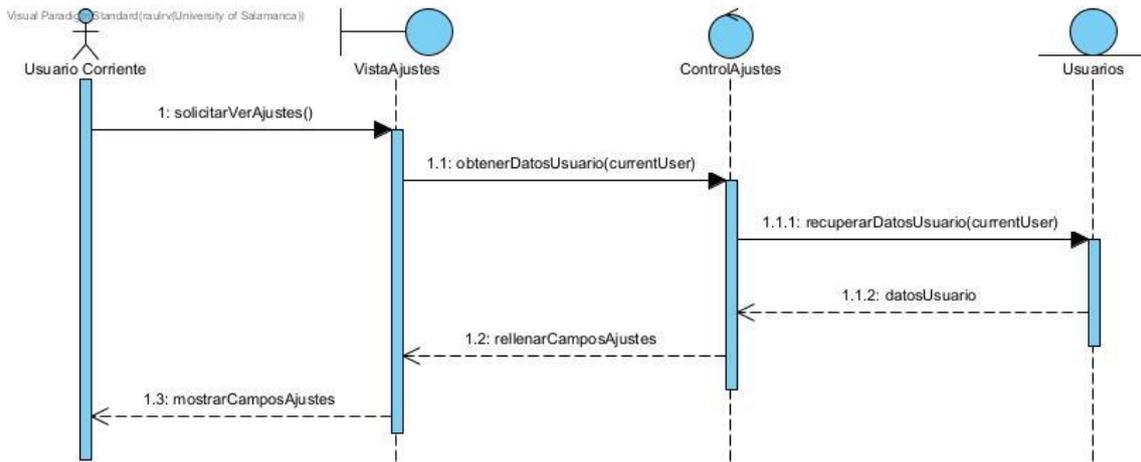


Imagen 24. UC-023 Ver ajustes

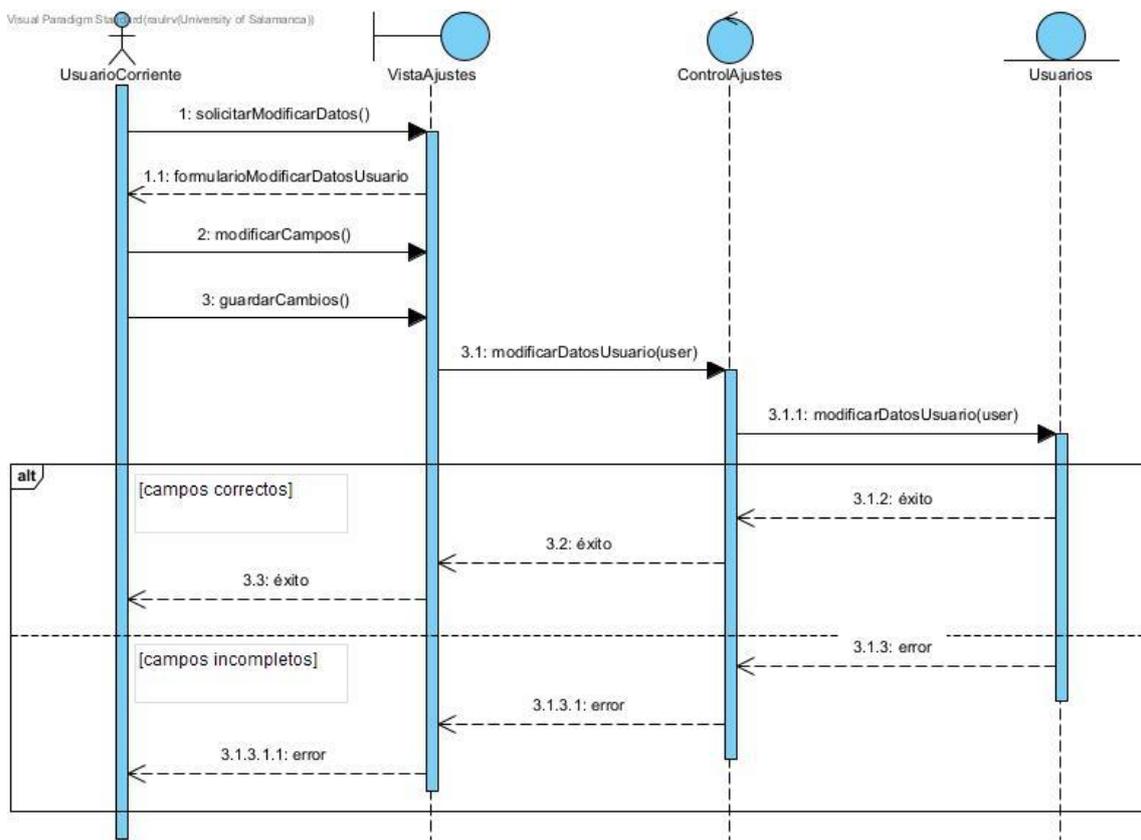


Imagen 25. UC-024 Modificar datos del usuario

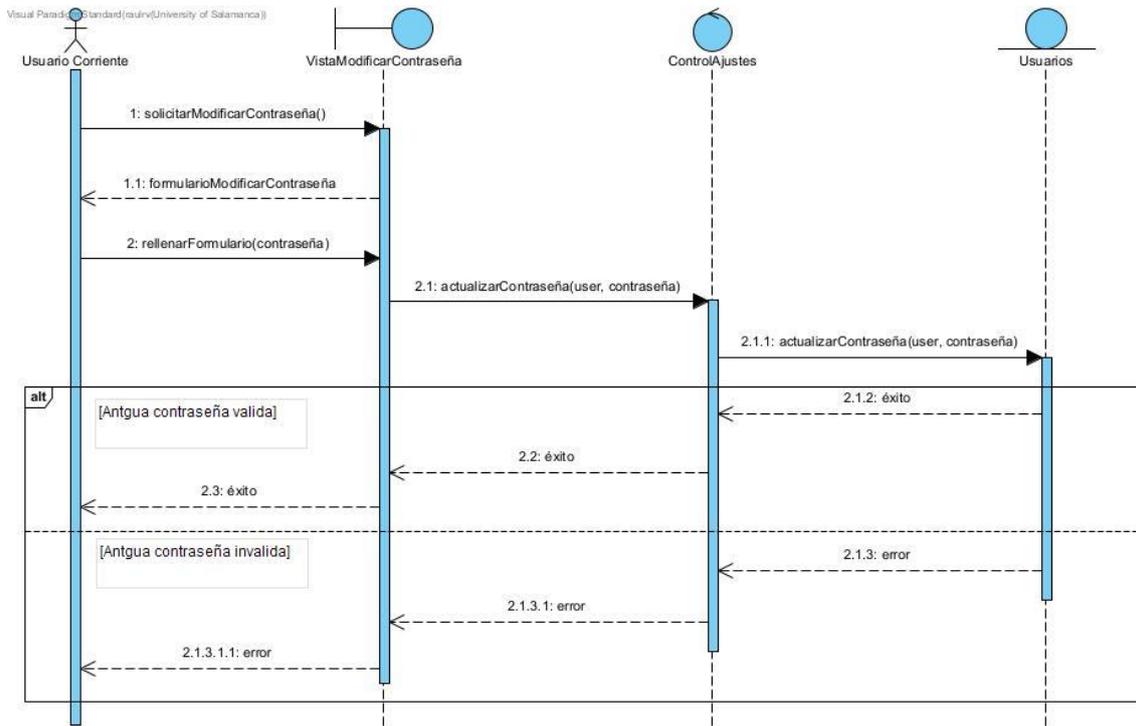


Imagen 26. UC-025 Modificar contraseña

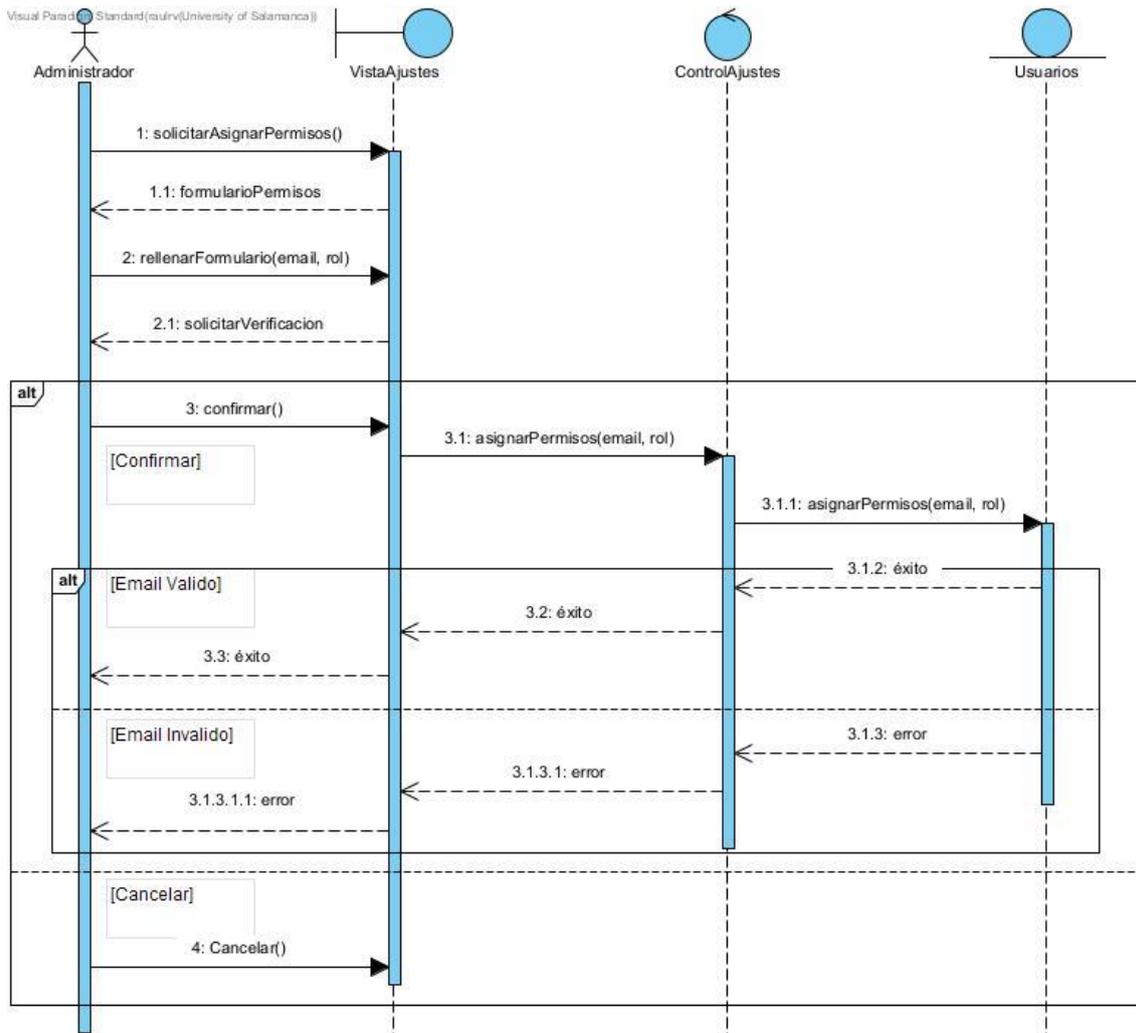


Imagen 27. UC-026 Asignar permisos a usuario

## 4. PROPUESTA DE ARQUITECTURA

En nuestra propuesta de arquitectura, hemos utilizado la estructura Modelo-Vista-Controlador (MVC) para organizar nuestro sistema. Con este enfoque de arquitectura conseguimos separar la lógica de negocio, la presentación de datos y la interacción del usuario con nuestra aplicación.

El patrón MVC consta de:

- Modelo: Aquí se encuentra la estructura de los datos, para los cuales hemos creado una serie de clases. También se crearán colecciones dentro de la base de datos que servirán de modelo, con las cuales interactuará el controlador.
- Vista: Es la parte gráfica, con la que el usuario está en contacto. En ella se presentarán los diferentes datos en función de las acciones realizadas por el usuario.
- Controlador: Es el intermediario entre el Modelo y la Vista, recibiendo las acciones proporcionadas por el usuario a la vista y actualizando el modelo, o recopilando datos del modelo para devolvérselos a la vista.

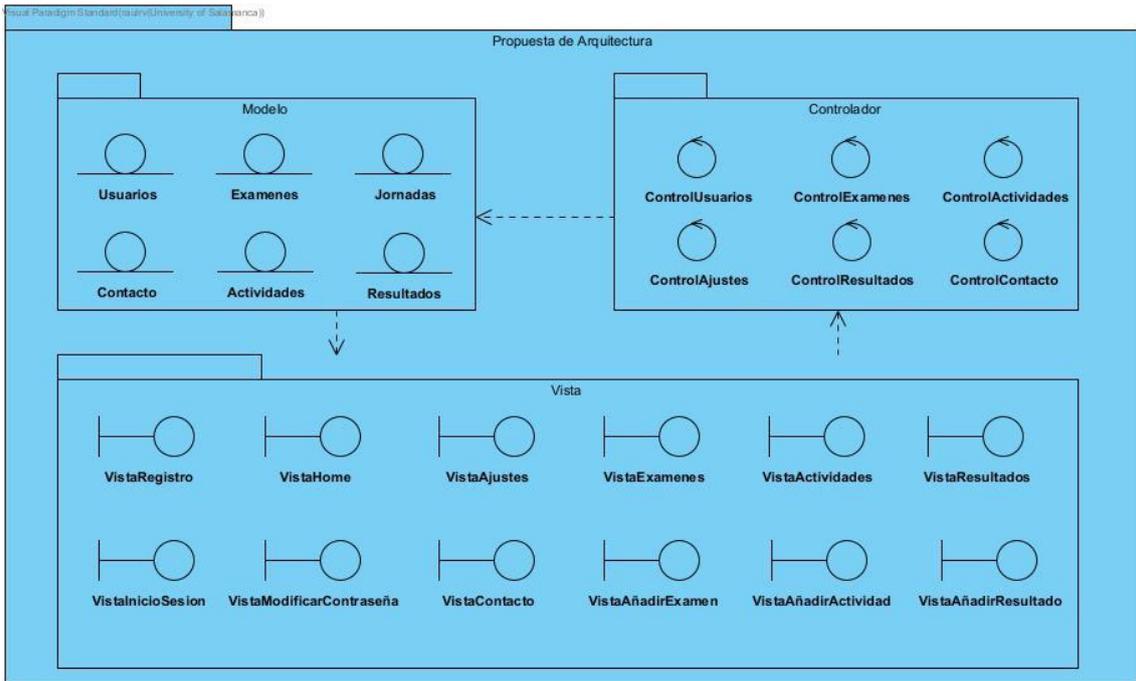


Imagen 28. Propuesta de arquitectura

## 5. REFERENCIAS

Moreno García M. N., Z. D. (2021). Transparencias de Ingeniería del Software I.

Moreno García M. N., Z. D. (2022). Transparencias de Gestión de Proyectos.

Moreno García M. N., Z. D.-A. (2022). Transparencias de Ingeniería del Software II.