



UNIVERSIDAD DE CÓRDOBA

Ingeniería del Software, Conocimiento y Bases de Datos

GRADO DE INGENIERÍA INFORMÁTICA

INGENIERÍA DE SOFTWARE

ENTREGA FINAL

AUTOR/ ES: GRUPO 29

Alicia Zamora Martín

Adrián López Ortiz

Gabriel Rodríguez Raya

Javier Luna Carmona

ÍNDICE DE CONTENIDOS

1.DESCRIPCIÓN DEL PROBLEMA	4
2.ESPECIFICACIÓN Y ANÁLISIS DE REQUISITOS	5
2.1 Requisitos funcionales.....	5
2.2 Requisitos no funcionales	6
3. CASOS DE USO.DIAGRAMA DE CASOS DE USO	7
C.U 01:Dar de alta a un paciente dentro del sistema.....	7
C.U 02: Buscar a un paciente dentro del sistema.....	8
C.U 03: Generar lista de pacientes.....	9
C.U 04: Modificar los datos personales de un paciente.....	10
C.U 05: Eliminar a un paciente.....	11
C.U 06: Consultar las citas de un paciente.....	12
C.U 07: Añadir una cita nueva para un paciente.....	13
C.U 08: Modificar cita.....	14
C.U 09: Cancelar cita.....	15
C.U 10: Ver el historial clínico de un paciente.....	16
C.U 11: Añadir información al historial clínico.....	17
C.U 12: Visualizar tratamiento de un paciente.....	18
C.U 13: Añadir un tratamiento nuevo para un paciente.....	19
C.U 14: Cancelar tratamiento.....	20
C.U 15: Visualizar citas del día actual.....	21
Diagrama de casos de uso.....	22
4. HISTORIAS DE USUARIO.....	23-30
5. CLASES. ESPECIFICACIÓN Y DIAGRAMA DE CLASES.....	31-33
6. DIAGRAMAS DE SECUENCIA.....	34-43
7. MATRICES DE VALIDACIÓN.....	44
8. PROCEDIMIENTO SCRUM.....	45-46

ÍNDICE DE TABLAS Y FIGURAS

Figura 3.1: Diagrama de casos de uso.....	23
Tabla 5.1: Clase paciente.....	33
Tabla 5.2: Clase cita.....	33
Tabla 5.3: Clase historial.....	34
Tabla 5.4: Clase tratamiento.....	34
Figura 5.1: Diagrama de clases.....	35
Figura 6.1: Dar de alta paciente.....	36
Figura 6.2: Buscar paciente.....	37
Figura 6.3: Generar lista de pacientes.....	37
Figura 6.4: Modificar paciente.....	38
Figura 6.5: Eliminar paciente.....	38
Figura 6.6: Consultar cita de un paciente.....	39
Figura 6.7: Añadir cita.....	39
Figura 6.8: Modificar cita.....	40
Figura 6.9: Cancelar cita.....	40
Figura 6.10: Visualizar historial clínico.....	41
Figura 6.11: Añadir información historial clínico.....	41
Figura 6.12: Visualizar tratamientos.....	42
Figura 6.13: Añadir tratamiento.....	42
Figura 6.14: Cancelar tratamiento.....	43
Figura 6.15: Visualizar citas día actual.....	43

1. Descripción del problema.

Se requiere el desarrollo de un software de gestión para una clínica médica. Este software será utilizado por la secretaría de la clínica y los doctores para registrar los datos relativos a los pacientes, las citas, el tratamiento y el historial médico pasado. Para ello, deberemos tener en cuenta una serie de pautas y requisitos para cumplir las expectativas que desea el profesorado.

2. Especificación y análisis de requisitos.

Las partes interesadas dentro de nuestro sistema van a ser el desarrollador y el administrador, los cuales van a hacer uso de los datos que gestiona el sistema:

Paciente: Nombre, apellidos, teléfono, dirección postal, fecha de nacimiento, sanidad pública o privada.

Cita: Fecha, hora, paciente.

Historial: Paciente, fecha, motivo de consulta.

Tratamiento: Paciente, nombre, fecha de comienzo del tratamiento, fecha de finalización, periodicidad, dosis, historial de tratamientos pasados.

1.1 Requisitos Funcionales (ordenados por prioridad)

RF1: Dar de alta un paciente.

RF2: Buscar paciente.

RF3: Generar lista de pacientes.

RF4: Modificar datos paciente.

RF5: Eliminar paciente.

- RF6: Consultar citas de un paciente.
- RF7: Añadir nueva cita.
- RF8: Modificar cita.
- RF9: Cancelar cita.
- RF10: Ver historial clínico del paciente.
- RF11: Añadir información al historial clínico.
- RF12: Visualizar tratamientos.
- RF13: Añadir tratamiento.
- RF14: Cancelar tratamiento
- RF15: Visualizar citas día actual.

1.2 Requisitos No funcionales

- RNF1: Sistema operativo Linux.
- RNF2: Interfaz human friendly.
- RNF3: Lenguaje de implementación: C++.
- RNF4: Confirmación de eliminación paciente

3. Casos de uso. Diagrama de casos de uso.

El único actor identificado dentro de nuestro sistema va a ser el administrador, que llevará a cabo todas las funcionalidades del sistema

Los distintos casos de uso nos van a permitir describir una actividad o tarea que realice el sistema; por lo que nos vamos a valer de ellos para describir nuestros requisitos.

Caso de uso 01: Dar de alta un paciente dentro del sistema

ID: 01. Descripción: Se introducen los datos del paciente y el sistema lo almacena.

Actores principales: Administrador.

Precondiciones:

- Qué el paciente no exista en la base de datos del sistema.

Flujo principal:

1. El administrador desea ingresar en el sistema a un nuevo paciente.
2. El administrador abre el cuadro de diálogo de dar de alta a un nuevo paciente.
3. El administrador introduce los datos del paciente, véase, nombre completo, teléfono de contacto, dirección postal, fecha nacimiento y pertenencia a sanidad pública o privada.
4. El sistema comprueba que el paciente no exista en la base de datos.
5. El sistema almacena la información del paciente en la base de datos.
6. El sistema muestra por pantalla, que el paciente ha sido dado de alta en la base de datos.

Postcondiciones:

- Se muestran al administrador si la operación llevada a cabo ha tenido éxito o no.

Flujos alternativos: 5.a. Si ya existe el paciente, se muestra un mensaje de error. 5.b. Si los datos introducidos no son válidos, se muestra un mensaje de error.

Caso de uso 02: Buscar a un paciente en el sistema

ID: 02. Descripción: Se introduce uno de los datos personales de un paciente y se busca su ficha en la base de datos.

Actores principales: Administrador.

Precondiciones:

- Qué el paciente ya exista en la base de datos del sistema.

Flujo principal:

1. El administrador desea buscar a un paciente en el sistema.
2. El administrador abre el cuadro de diálogo de buscar a un paciente.
3. El administrador introduce uno de los datos del paciente.
4. El sistema busca al paciente en la base de datos.
5. El sistema muestra la información del paciente.
6. El sistema muestra un cuadro de acción, en el que están las opciones; ver el historial clínico del paciente, añadir una nueva cita, modificar una cita, cancelar una cita y volver al menú inicial.

Postcondiciones:

- Ninguna.

Flujos alternativos: 5.a. Si no existe el paciente, se muestra un mensaje de error.

Caso de uso 03: Generar lista de pacientes

ID: 03. Descripción: Se muestra por pantalla la lista de pacientes.

Actores principales: Administrador.

Precondiciones:

- Qué haya al menos un paciente.

Flujo principal:

1. El administrador desea generar una lista de pacientes.
2. El administrador abre el cuadro de diálogo de generar la lista de pacientes de la clínica.
3. El sistema muestra por pantalla una lista, con el nombre de cada uno de los pacientes.

Postcondiciones:

- Ninguna.

Flujos alternativos: 3.a. Si hay ningún paciente en la base de datos, se muestra un mensaje de error, diciendo que no hay ningún paciente en el sistema.

Caso de uso 04: Modificar los datos personales de un paciente

ID: 04. Descripción: Se modifican los datos personales del paciente.

Actores principales: Administrador.

Precondiciones:

- Qué el paciente exista en la base de datos del sistema.

Flujo principal:

1. El administrador desea modificar a un paciente de la base de datos del sistema.
2. El administrador busca y localiza al paciente cuyos datos se van a modificar (caso de uso <02>).
3. El administrador introduce los nuevos datos del paciente, véase, nombre completo, teléfono de contacto, dirección postal, fecha nacimiento y pertenencia a sanidad pública o privada.
4. El sistema comprueba que el paciente modificado tenga los mismos datos que otro paciente de la base de datos.
5. El sistema muestra por pantalla los nuevos datos del paciente.
6. El administrador, mediante un cuadro de diálogo, le da el visto bueno a la modificación.
7. El sistema almacena la información del paciente en la base de datos.
8. El sistema muestra por pantalla, que el paciente ha sido modificado.

Postcondiciones:

- Que se haya efectuado el cambio correctamente.

Flujos alternativos: 2.a. Si el paciente no existe, se muestra un mensaje de error. 5.a. Si los datos introducidos no son válidos, se muestra un mensaje de error. 5.b. Si el paciente modificado, coincide con otro paciente en cuanto a datos se refiere, se muestra un mensaje de error. 7.a. Si el administrador no confirma los cambios, los datos del paciente no se modifican.

Caso de uso 05: Eliminar a un paciente

ID: 05. Descripción: Se elimina a un paciente de la base de datos.

Actores principales: Administrador.

Precondiciones:

- Qué el paciente exista en la base de datos del sistema.

Flujo principal:

1. El administrador desea eliminar a un paciente de la base de datos del sistema.
2. El administrador abre el cuadro de diálogo de eliminar a un paciente de la base de datos.
3. El administrador busca y localiza al paciente cuyos datos de van a eliminar (caso de uso <02>).
4. El administrador confirma en el cuadro de diálogo entrante de qué quiere eliminar.
5. El sistema borra de la base de datos del propio sistema al paciente.

Postcondiciones:

- Que se haya efectuado el cambio correctamente.
- Que no se encuentre el paciente en la base de datos.

Flujos alternativos: 4.a. Si el paciente no existe, se muestra un mensaje de error. 5.a. Si el administrador cancela la eliminación de dicho paciente de la base de datos, se vuelve al menú principal.

Caso de uso 06: Consultar las citas de un paciente

ID: 06. Descripción: El sistema muestra las citas de un paciente.

Actores principales: Administrador.

Precondiciones:

- Que el paciente exista en la base de datos.

Flujo principal:

1. El administrador desea mostrar las citas de un paciente en la pantalla.
2. El administrador abre el cuadro de diálogo de ver el historial de citas de un paciente.
3. El administrador busca y localiza al paciente cuyos datos se van a modificar (caso de uso <02>).
4. El administrador abre el cuadro de diálogo de consultar historial de citas de un paciente.
5. El sistema muestra por pantalla los datos personales del paciente (nombre completo, teléfono, dirección postal, fecha de nacimiento y tipo de sanidad contratada) junto con todas las citas de este.

Postcondiciones:

- Ninguna.

Flujos alternativos: 5.a. Si el paciente no ha tenido una cita todavía, se mostrará un mensaje aclarándolo.

Caso de uso 07: Añadir una cita nueva para un paciente

ID: 07. Descripción: Se introducen los datos de la nueva cita para el paciente y el sistema lo almacena.

Actores principales: Administrador.

Precondiciones:

- Qué el paciente al que se quiera dar una nueva cita pertenezca a la base de datos.
- Qué el no haya una cita previa pendiente para el paciente.

Flujo principal:

1. El administrador desea ingresar en el sistema una nueva cita para un paciente determinado.
2. El administrador abre el cuadro de diálogo de añadir nueva cita.
3. El administrador busca y localiza al paciente cuyos datos se van a modificar (caso de uso <02>).
4. El administrador introduce los datos de la nueva cita que se le quiere asignar al paciente.
5. El sistema comprueba que no exista ya otra cita en la fecha y hora indicados.
6. El sistema almacena la información de la nueva cita en la base de datos.
7. El sistema muestra por pantalla, que el paciente tiene una nueva cita.

Postcondiciones:

- Se muestran al administrador si la operación llevada a cabo ha tenido éxito o no.

Flujos alternativos: 4.a. Si no existe el paciente, se muestra un mensaje de error. 6.a. Si ya existe una cita con esa fecha y hora, se muestra un mensaje de error.

Caso de uso 08: Modificar cita

ID: 08. Descripción: El sistema modifica una cita existente.

Actores principales: Administrador.

Precondiciones:

- Que la cita exista en la base de datos.

Flujo principal:

1. El administrador desea modificar una cita de la base de datos del sistema.
2. El administrador busca y localiza al paciente cuya cita se va a modificar (caso de uso <02>).
3. El administrador abre el cuadro de diálogo de modificar cita.
4. El administrador introduce los nuevos datos de la cita, véase: Nombre completo del paciente, fecha y hora.
5. El sistema comprueba que la cita modificada no tenga la misma fecha que otra cita de la base de datos.
6. El sistema muestra por pantalla los nuevos datos de la cita.
7. El administrador, mediante un cuadro de diálogo, le da el visto bueno a la modificación.
8. El sistema almacena la información de la cita en la base de datos.
9. El sistema muestra por pantalla, que la cita ha sido modificada.

Postcondiciones:

- Se muestran al administrador si la operación llevada a cabo ha tenido éxito o no.

Flujos alternativos: 6.a. Si no modifica ningún campo, se mostrará una advertencia. 6.b. Si los datos introducidos no son válidos, se muestra un mensaje de error.

Caso de uso 09: Cancelar cita

ID: 09. Descripción: El sistema cancela una cita existente.

Actores principales: Administrador.

Precondiciones:

- Que la cita exista en la base de datos.

Flujo principal:

1. El administrador desea cancelar una cita de la base de datos del sistema.
2. El administrador abre el cuadro de diálogo de cancelar una cita de la base de datos.
3. El administrador busca y localiza al paciente cuya cita se va a eliminar (caso de uso <02>).
4. El administrador abre el cuadro de diálogo de eliminar cita.
5. El sistema muestra por pantalla los datos personales de la cita: Fecha, hora y nombre completo del paciente.
6. El administrador confirma en el cuadro de diálogo entrante que quiere eliminar la cita.
7. El sistema borra de la base de datos del propio sistema la cita.

Postcondiciones:

- Que se haya efectuado el cambio correctamente.
- Que no se encuentre la cita en la base de datos.

Flujos alternativos: 5.a Si no hay una próxima cita en la base de datos, muestra un mensaje de error. 7.a. Si el administrador cancela la eliminación de dicha cita de la base de datos, se vuelve al menú principal.

Caso de uso 10: Ver el historial clínico de un paciente

ID: 10. Descripción: El sistema muestra el historial clínico de un paciente.

Actores principales: Administrador.

Precondiciones:

- Que el paciente exista en la base de datos.

Flujo principal:

1. El administrador desea mostrar el historial clínico de un paciente en la pantalla.
2. El administrador abre el cuadro de diálogo de ver el historial clínico de un paciente.
3. El administrador busca y localiza al paciente cuyos datos se van a modificar (caso de uso <02>).
4. El administrador abre el cuadro de diálogo de consultar el historial clínico de un paciente.
5. El sistema muestra por pantalla los datos personales del paciente (nombre completo, teléfono, dirección postal, fecha de nacimiento y tipo de sanidad contratada), así como las citas previas que ha realizado, con sus correspondientes datos y los tratamientos que han sido recetados.

Postcondiciones:

- Ninguna.

Flujos alternativos: 5.a. Si el paciente no ha tenido una cita todavía, se mostrará un mensaje aclarándolo.

Caso de uso 11: Añadir información al historial clínico

ID: 11. Descripción: Poder añadir nueva información relativa al estado de salud y enfermedades del paciente.

Actores principales: Administrador.

Precondiciones:

- Que exista el paciente
- Que exista un historial médico ya creado

Flujo principal:

1. El administrador desea añadir nueva información al historial clínico.
2. El administrador busca y localiza al paciente cuyo historial se va a visualizar (caso de uso <02>).
3. Una vez en el paciente se selecciona la opción añadir información al historial clínico, se procede y se guarda la nueva la nueva información.

Postcondiciones:

- Que se halla añadido correctamente la nueva información.

Flujos alternativos: 3. Ninguno

Caso de uso 12: Visualizar tratamiento de un paciente

ID: 12. Descripción: Se visualiza el tratamiento de un paciente.

Actores principales: Administrador.

Precondiciones:

- Qué el paciente exista en la base de datos del sistema.

Flujo principal:

1. El administrador desea visualizar el tratamiento de un paciente de la base de datos del sistema.
2. El administrador busca y localiza al paciente cuyo tratamiento se va a visualizar (caso de uso <02>).
3. El sistema muestra por pantalla el tratamiento del paciente.

Postcondiciones:

- Ninguno.

Flujos alternativos: 2.a. Si el paciente no existe, se muestra un mensaje de error.

Caso de uso 13: Añadir un tratamiento nuevo para un paciente

ID: 13. Descripción: Se introducen los datos del nuevo tratamiento para el paciente y el sistema lo almacena.

Actores principales: Administrador.

Precondiciones:

- Qué el paciente pertenezca a la base de datos.

Flujo principal:

1. El administrador desea ingresar en el sistema un nuevo tratamiento para un paciente determinado.
2. El administrador abre el cuadro de diálogo de añadir nuevo tratamiento.
3. El administrador busca y localiza al paciente cuyos datos se van a modificar (caso de uso <02>).
4. El administrador introduce los datos del nuevo tratamiento que se le quiere asignar al paciente.
5. El sistema comprueba que no exista ya otro tratamiento igual.
6. El sistema almacena la información del nuevo tratamiento en la base de datos.
7. El sistema muestra por pantalla, que el paciente tiene un nuevo tratamiento.

Postcondiciones:

- Se muestran al administrador si la operación llevada a cabo ha tenido éxito o no.

Flujos alternativos: 4.a. Si no existe el paciente, se muestra un mensaje de error. 6.a. Si ya existe dicho tratamiento, se muestra un mensaje de error.

Caso de uso 14: Cancelar tratamiento

ID: 14. Descripción: El sistema cancela un tratamiento existente.

Actores principales: Administrador.

Precondiciones:

- Que el tratamiento exista en la base de datos.

Flujo principal:

1. El administrador desea cancelar un tratamiento de un paciente.
2. El administrador abre el cuadro de diálogo de cancelar una cita de la base de datos.
3. El administrador busca y localiza al paciente cuyo tratamiento se va a cancelar (caso de uso <02>).
4. El administrador abre el cuadro de diálogo de cancelar tratamiento.
5. El sistema muestra por pantalla los datos personales del tratamiento: nombre del tratamiento, fecha de inicio, fecha de finalización, periodicidad, dosis e historial de tratamientos pasados.
6. El administrador confirma en el cuadro de diálogo entrante que quiere cancelar el tratamiento.
7. El sistema marca como cancelado del propio sistema el tratamiento.

Postcondiciones:

- Que se haya efectuado el cambio correctamente.
- Que no se encuentre la cita en la base de datos.

Flujos alternativos: 5.a Si no hay un tratamiento en la base de datos, muestra un mensaje de error. 7.a. Si el administrador anula la cancelación de dicho tratamiento, se vuelve al menú principal.

Caso de uso 15: Visualizar las citas del día actual

ID: 15. Descripción: El sistema muestra las citas del día actual.

Actores principales: Administrador.

Precondiciones:

- Ninguna.

Flujo principal:

1. El administrador desea mostrar las citas del día actual en la pantalla.
2. El administrador abre el cuadro de diálogo de ver las citas del día actual.
3. El sistema muestra por pantalla las citas del día actual, con sus correspondientes datos: Nombre del paciente, fecha y hora.

Postcondiciones:

- Ninguna.

Flujos alternativos: 3.a. Si no existe ninguna cita en el día actual, se mostrará un mensaje aclarándolo.

Los **diagramas de casos de uso** sirven para especificar la comunicación y el comportamiento de un sistema mediante su interacción con los usuarios y/u otros sistemas. O lo que es igual, un **diagrama** que muestra la relación entre los actores y los **casos de uso** en un sistema.

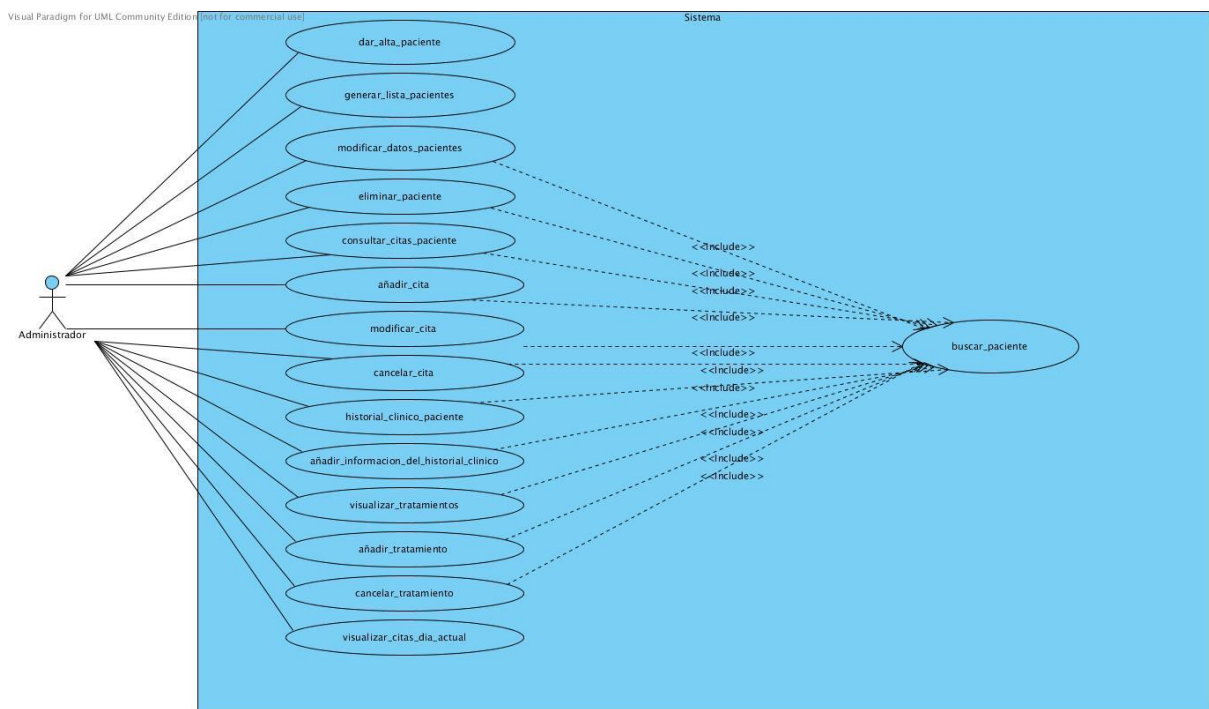


Figura 3.1:Diagrama de casos de uso

4. Historias de usuario

ID: 01. **Nombre:** Dar de alta a un paciente en el sistema.

Prioridad (de 1 a 15): 1. **Puntos estimado:** 2. **Iteración:** 2.

Responsable: Adrián López Ortiz.

Descripción:

Como *administrador* quiero *poder dar de alta a un paciente* para *tenerlo dentro del sistema y poder guardar información personal y clínica sobre ese paciente*.

Validación:

- Se debe poder dar de alta a cualquier paciente que no esté dentro del sistema.
- Se debe introducir la información personal de cada paciente para poder darlo de alta.
- Es necesario utilizar el nombre completo.
- Una vez dado de alta el paciente ha de aparecer en la base de datos.
- Se debe poder concertar una cita para el paciente dado de alta.
- Se debe poder incorporar un historial médico.
- Se debe poder incorporar información sobre tratamientos para el paciente.

ID: 02. Nombre: Buscar paciente en el sistema.

Prioridad (de 1 a 15): 2. **Puntos estimado:** 2. **Iteración:** 2.

Responsable: Adrián López Ortiz.

Descripción:

Como *administrador* quiero *localizar a un paciente* para *consultar su información personal e historial médico*.

Validación:

- Se debe poder localizar cualquier paciente.
- Los resultados de la búsqueda han de ofrecer un solo resultado.
- Se deben mostrar todos los datos del paciente.
- Se debe mostrar el historial médico del paciente.
- Se deben mostrar los tratamientos que el paciente ha recibido.

ID: 03. Nombre: Generar una lista de los pacientes dentro del sistema.

Prioridad (de 1 a 15): 3. **Puntos estimado:** 1. **Iteración:** 3.

Responsable: Alicia Zamora Martín.

Descripción:

Como *administrador* quiero *poder generar una lista de pacientes* para *ver los pacientes que hay dentro del sistema*.

Validación:

- Deben aparecer todos los pacientes del sistema.
- Se debe mostrar su nombre completo.

ID: 04. **Nombre:** Modificar datos del paciente.

Prioridad (de 1 a 15): 4. **Puntos estimado:** 4. **Iteración:** 2.

Responsable: Javier Luna Carmona.

Descripción:

Como *administrador* quiero *poder modificar los datos de los pacientes* para *tener la información actualizada*.

Validación:

- Se deben poder actualizar los datos de cualquier paciente que esté dado de alta en el sistema.
- Se deben eliminar los datos obsoletos de cada paciente.
- La nueva información sobre el paciente ha de quedar guardada correctamente dentro del sistema.

ID: 05. **Nombre:** Eliminar paciente dentro del sistema.

Prioridad (de 1 a 15): 5. **Puntos estimado:** 2. **Iteración:** 2.

Responsable: Gabriel Rodríguez Raya.

Descripción:

Como *administrador* quiero *poder eliminar a un paciente* para *tener un sistema actualizado*.

Validación:

- Se debe eliminar cualquier registro personal del usuario, incluido el historial clínico.
- Se debe eliminar el paciente de la base de datos.

ID: 06. **Nombre:** Consultar las citas de un paciente perteneciente al sistema.

Prioridad (de 1 a 15): 6. **Puntos estimado:** 3. **Iteración:** 3.

Responsable: Gabriel Rodríguez Raya.

Descripción:

Como *administrador* quiero *poder buscar las citas de un paciente* para *tener dicha información*.

Validación:

- Se debe eliminar cualquier registro personal del usuario, incluido el historial clínico.
- Se debe eliminar el paciente de la base de datos.

ID: 07. **Nombre:** Añadir cita.

Prioridad (de 1 a 15): 7. **Puntos estimado:** 3. **Iteración:** 1.

Responsable: Adrián López Ortiz.

Descripción:

Como *administrador* quiero *poder añadir una cita* para *poder dar hora de consulta a un paciente*.

Validación:

- Se debe poder añadir una cita para cualquier paciente dado de alta dentro del sistema.
- Se debe establecer una fecha.
- Se debe establecer una hora.
- Debe aparecer el nombre completo del paciente.
- Se debe comprobar que no hay una cita establecida con anterioridad que tenga la fecha y hora deseadas para poder añadir la nueva cita.

ID: 08.

Nombre: Modificar cita.

Prioridad (de 1 a 15): 8.

Puntos estimado: 4.

Iteración: 1.

Responsable: Gabriel Rodríguez Raya.

Descripción:

Como *administrador* quiero *poder modificar una cita* para *poder cambiar la fecha y/o hora de la misma*.

Validación:

- Se debe haber añadido esa cita con anterioridad para poder modificarla.
- Se debe comprobar disponibilidad antes de poder modificarla.
- Se debe poder modificar la fecha.
- Se debe poder modificar la hora.
- Se deben guardar los cambios dentro del sistema.

ID: 09. **Nombre:** Cancelar cita.

Prioridad (de 1 a 15): 9. **Puntos estimado:** 2. **Iteración:** 1.

Responsable: Javier Luna Carmona.

Descripción:

Como *administrador* quiero *poder cancelar una cita* para *aquellos pacientes que no puedan asistir*

Validación:

- Se debe poder eliminar cualquier cita añadida con anterioridad que aún no haya pasado.
- Se debe eliminar totalmente del sistema.

ID: 10. **Nombre:** Ver historial clínico del paciente.

Prioridad (de 1 a 15): 10. **Puntos estimado:** 2. **Iteración:** 4.

Responsable: Javier Luna Carmona.

Descripción:

Como *administrador* quiero *poder ver el historial del paciente* para *poder conocer sus enfermedades*.

Validación:

- Se debe poder ver el historial clínico de cualquier paciente dado de alta.

ID: 11. **Nombre:** Añadir información al historial clínico.

Prioridad (de 1 a 15): 11. **Puntos estimado:** 4. **Iteración:** 4.

Responsable: Adrián López Ortiz.

Descripción:

Como *administrador* quiero *poder añadir información al historial del paciente* para *poder llevar un seguimiento de sus enfermedades*.

Validación:

- Se debe poder añadir información al historial clínico de cualquier paciente dado de alta.

ID: 12. Nombre: Visualizar tratamientos.

Prioridad (de 1 a 13): 12. **Puntos estimado:** 2. **Iteración:** 4.

Responsable: Gabriel Rodríguez Raya.

Descripción:

Como *administrador* quiero *poder visualizar los tratamientos del paciente para poder conocer cómo se están tratando sus enfermedades.*

Validación:

- Se debe poder visualizar los tratamientos de cualquier paciente dado de alta.

ID: 13. Nombre: Añadir tratamiento.

Prioridad (de 1 a 15): 13. **Puntos estimado:** 4. **Iteración:** 3.

Responsable: Gabriel Rodríguez Raya.

Descripción:

Como *administrador* quiero *poder añadir un tratamiento para poder realizar un seguimiento de los tratamientos que está realizando un paciente para tratar sus enfermedades.*

Validación:

- Se debe poder añadir un tratamiento para cualquier paciente dado de alta dentro del sistema.
- Se debe indicar el nombre del medicamento o tratamiento.
- Se debe indicar la dosis o periodo de duración.

ID: 14. Nombre: Cancelar tratamiento.

Prioridad (de 1 a 15): 14. **Puntos estimado:** 3. **Iteración:**3.

Responsable: Javier Luna Carmona.

Descripción:

Como *administrador* quiero *poder cancelar un tratamiento de un paciente para poder realizar un seguimiento de los tratamientos que está realizando un paciente para tratar sus enfermedades.*

Validación:

- Se debe poder cancelar un tratamiento para cualquier paciente dado de alta dentro del sistema.
- Se debe indicar el nombre del tratamiento.

ID: 15. Nombre: Visualizar citas del día actual.

Prioridad (de 1 a 15): 15. **Puntos estimado:** 4. **Iteración:** 4.

Responsable: Alicia Zamora Martín.

Descripción:

Como *administrador* quiero *poder visualizar las citas del día actual para conocer que pacientes vendrán a consulta ese día.*

Validación:

- Se deben visualizar todas las citas para el día actual.
- Se debe visualizar el nombre completo del paciente.
- Se debe visualizar la hora.
- Se debe comprobar que no aparezcan las citas canceladas.

5. Clases. Especificación y diagrama de clases.

Clase: Paciente
Descripción
Esta clase contendrá las funciones del paciente. Entre ellas se encuentra la de cargar un paciente en el sistema y la de borrar paciente.
Datos
-nombre (string): Nombre del paciente dentro del sistema
-apellidos (string) : Apellidos del paciente dentro del sistema
-teléfono (string) : Teléfono del paciente
-dirección postal (string) : Dirección postal del paciente a la que se le mandarán las citas
-fecha de nacimiento (string) : Fecha de nacimiento del paciente
-tarjeta sanitaria (boolean) : Adoptará valor true si el paciente viene de la seguridad pública y false si viene de la privada
Métodos
+cargarPaciente : Función mediante la cual se buscará a un paciente dentro de la base de datos del sistema
+guardarPaciente : Función mediante la cual se añadirán y guardarán los datos y actualizaciones de la información de un paciente dentro del sistema
+borrarPaciente : Función mediante la cual se busca a un paciente dentro de la base de datos del sistema y se elimina

Tabla 5.1: Clase paciente

Clase: Cita
Descripción
Esta clase contendrá las funciones e información referentes a las citas de los distintos pacientes.
Datos
-fecha(string) : Indica la fecha en la que la cita tendrá lugar
-hora (string) : Indica la hora en la que la cita tendrá lugar
Métodos
+cargarCita : Función mediante la cuál se buscará la cita de un paciente dentro de la base de datos del sistema
+guardarCita : Función mediante la cual se añadirán y guardarán los datos y actualizaciones de la cita de un paciente dentro del sistema
+borrarCita : Función mediante la cual se buscará una cita concreta mediante la fecha y se anulará.

Tabla 5.2: Clase cita

Clase: Registro de historial
Descripción
Esta clase contendrá las funciones e información referentes al historial clínico de los distintos pacientes.
Datos
-motivo de consulta (string) : Indica el motivo por el que el paciente ha acudido a la clínica
-fecha (string) : Indica la hora en la que la consulta ha tenido lugar
Métodos
+cargarHistorial : Función mediante la cuál se buscará el historial médico de un paciente dentro de la base de datos del sistema
+guardarHistorial : Función mediante la cual se añadirá y guardará información en el historial médico de uno de los pacientes dentro del sistema.
+borrarHistorial : Función mediante la cual se buscará el historial médico de uno de los pacientes contenidos en la base de datos del sistema y se eliminará.

Tabla 5.3: Clase historial

Clase: Tratamiento
Descripción
Esta clase contendrá las funciones e información referente a los distintos tratamientos asignados a los pacientes dentro del sistema.
Datos
-nombre (string): Nombre del tratamiento
-fecha de inicio (string) : fecha en la que el paciente comienza a tomar el tratamiento
-fecha de fin (string) : fecha en la que el paciente termina de tomar el tratamiento
-periodicidad (string) : Indica cada cuanto tiempo el paciente debe tomar el tratamiento asignado
-dosis(string) : Indica la dosis del tratamieto asignado que el paciente debe de tomar
Métodos
+cargarTratamiento : Función mediante la cual se buscará un tratamiento dentro de la base de datos del sistema
+guardarTratamiento : Función mediante la cual se añadirán nuevos tratamientos y se guardarán las actualizaciones que los tratamientos de un paciente dentro del sistema sufran
+borrarTratamiento : Función mediante la cual se buscará uno de los tratamientos de la base de datos del sistema y se eliminará.

Tabla 5.4: Clase tratamiento

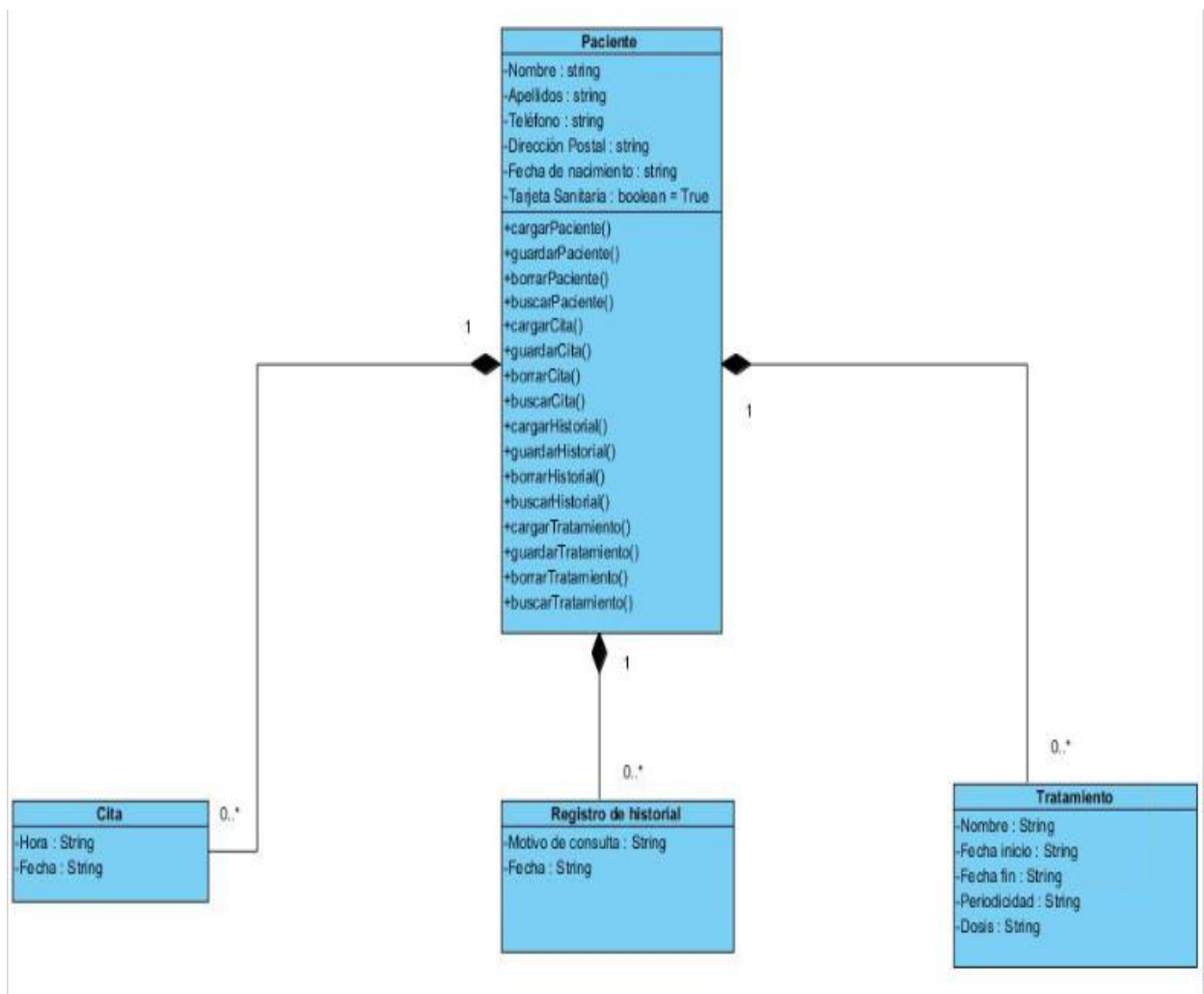


Figura 5.1: Diagrama de clases

6. Diagramas de secuencia

Un diagrama de secuencias muestra la interacción de un conjunto de objetos de una aplicación a través del tiempo, en el cual se indicarán los módulos o clases que formaran parte del programa y las llamadas que se hacen cada uno de ellos para realizar una tarea determinada, por esta razón permite observar la perspectiva cronológica de las interacciones. Es importante recordar que el diagrama de secuencias se realiza a partir de la descripción de un caso de uso.

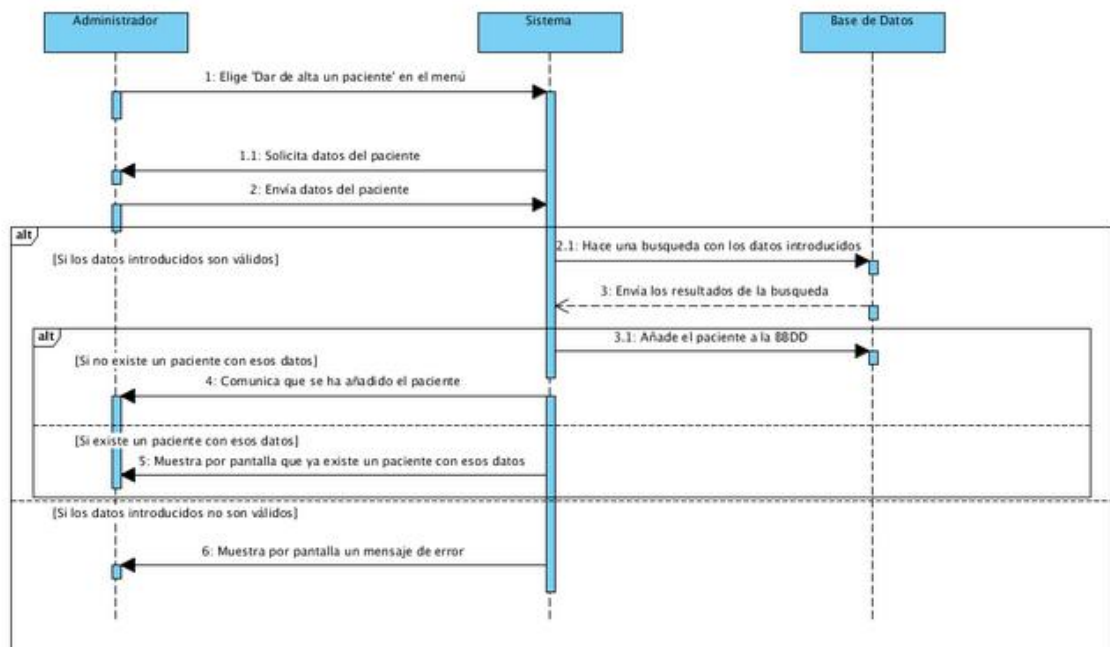


Figura 6.1: Dar de alta paciente

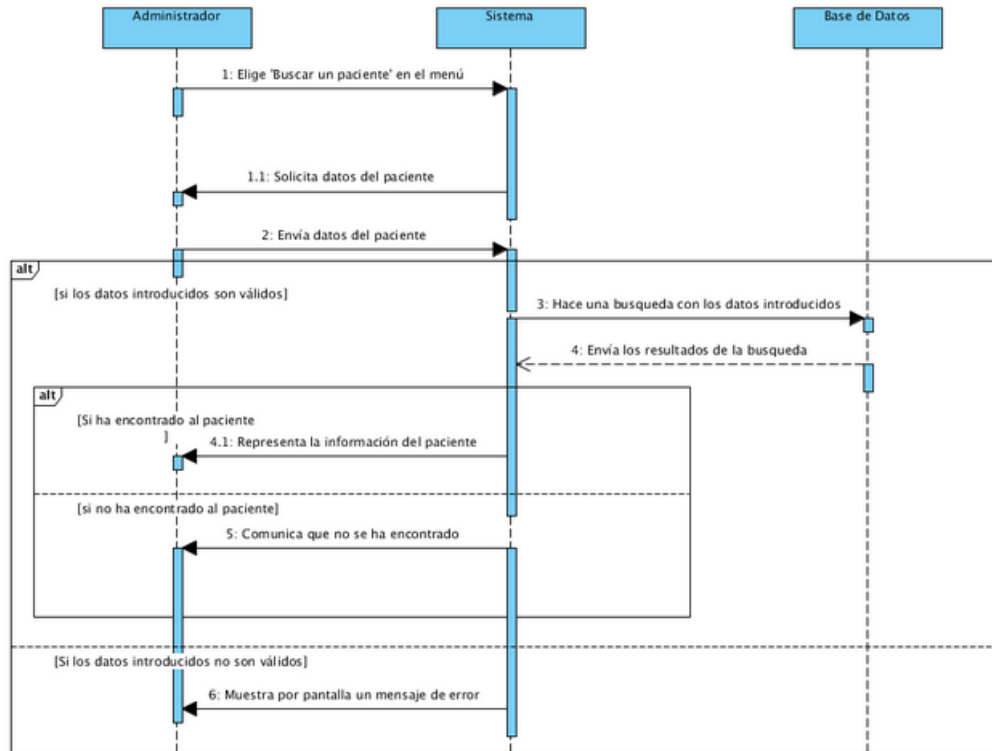


Figura 6.2: Buscar paciente

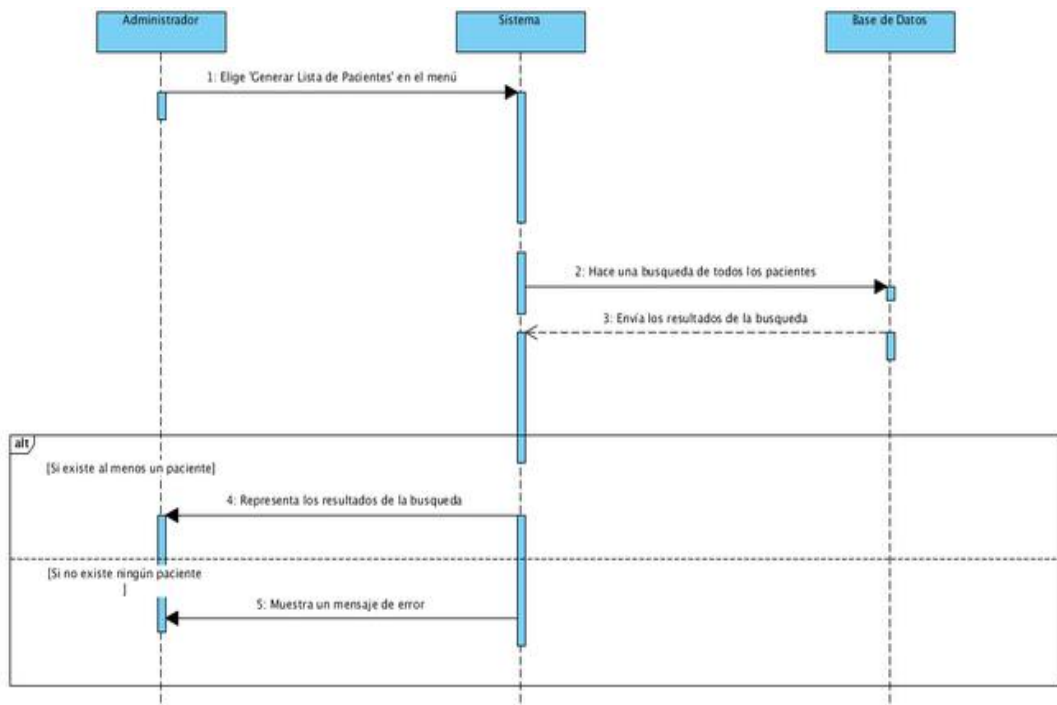


Figura 6.3: Generar lista pacientes

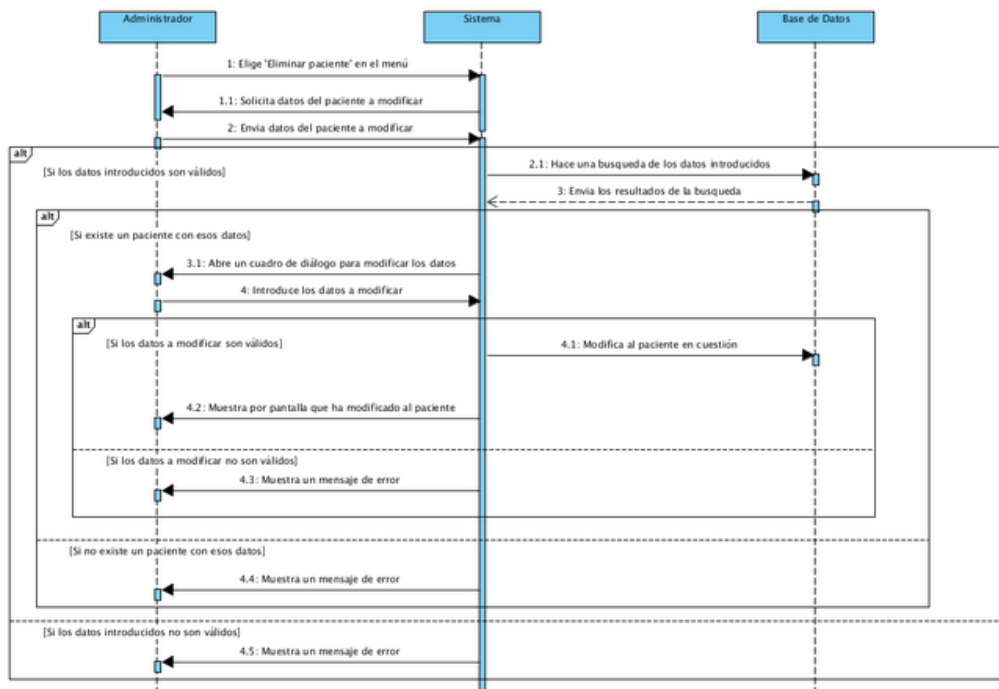


Figura 6.4: Modificar datos paciente

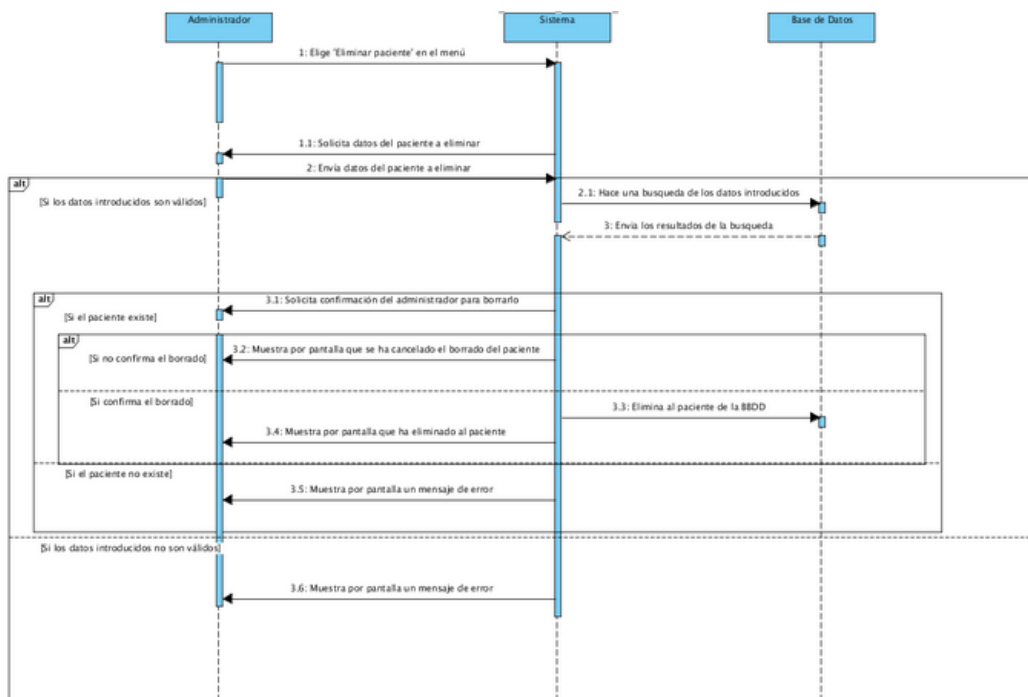


Figura 6.5: Eliminar paciente

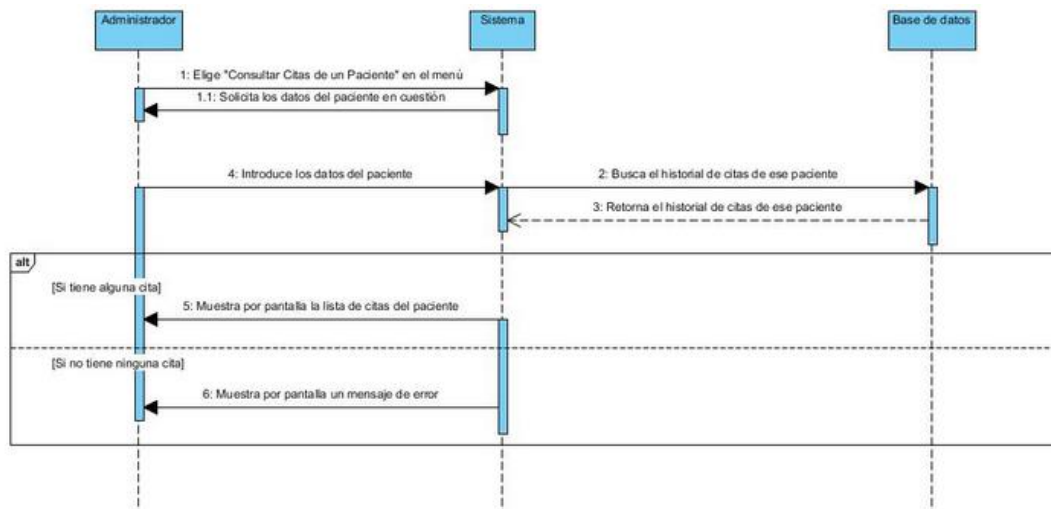


Figura 6.6: Consultar cita de un paciente

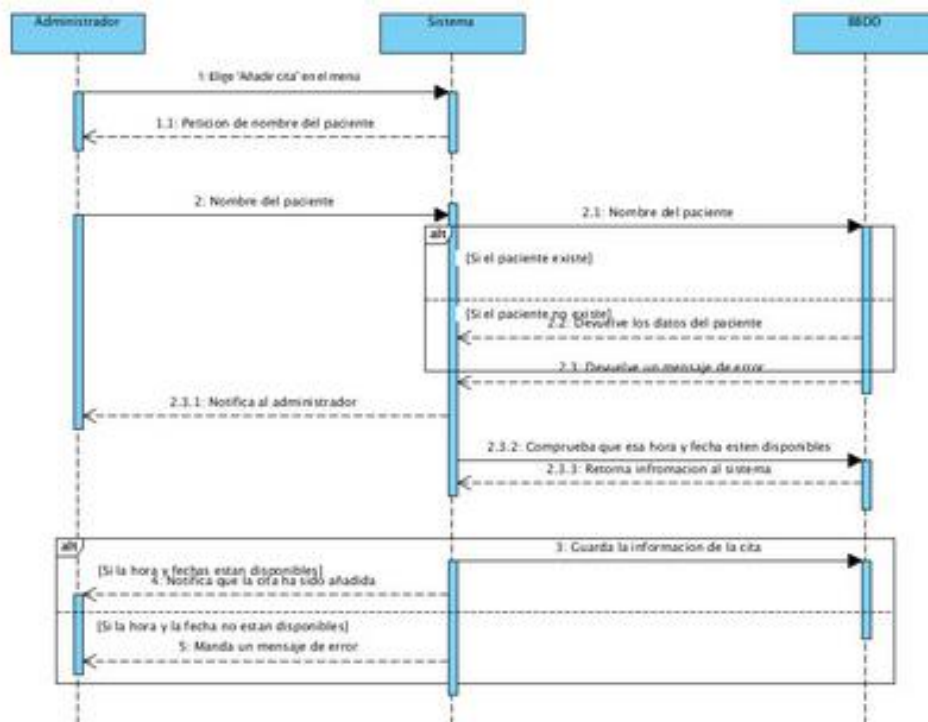


Figura 6.7: Añadir cita

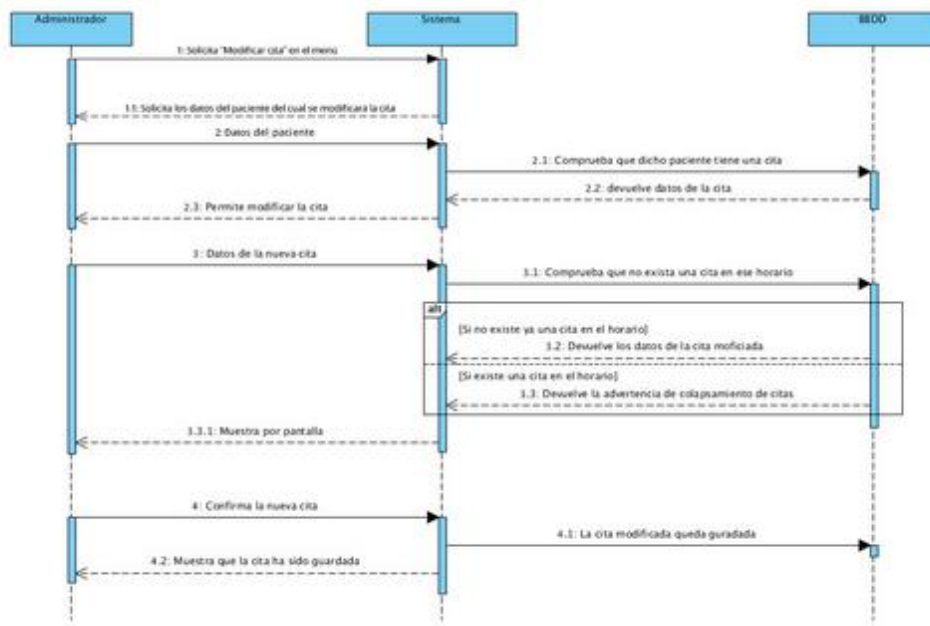


Figura 6.8: Modificar cita

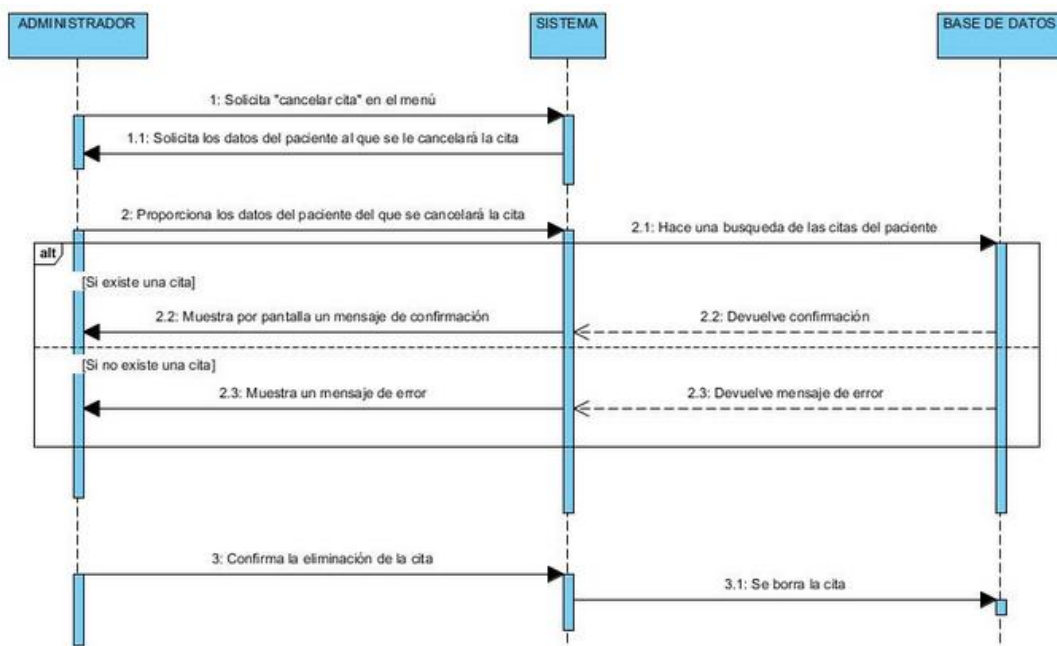


Figura 6.9: Cancelar cita

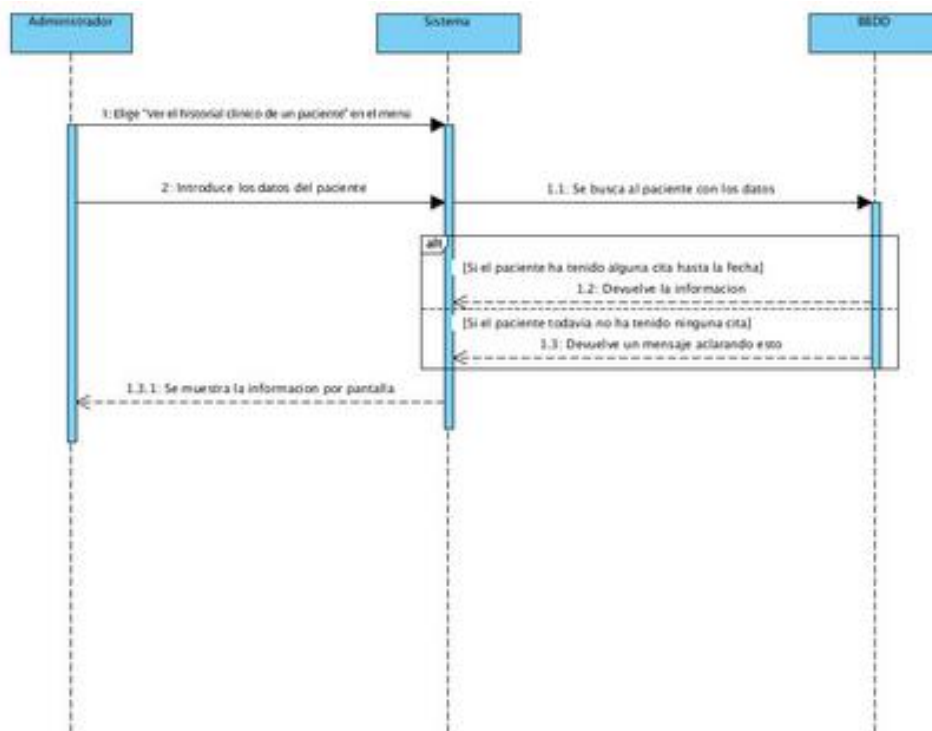


Figura 6.10: Visualizar historial clínico

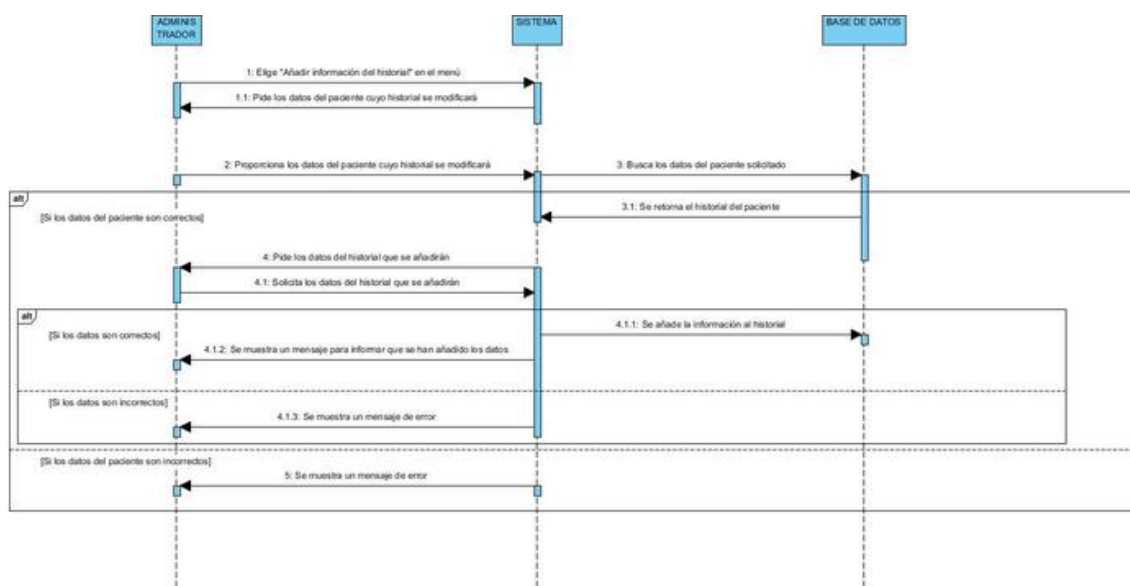


Figura 6.11: Añadir información historial clínico

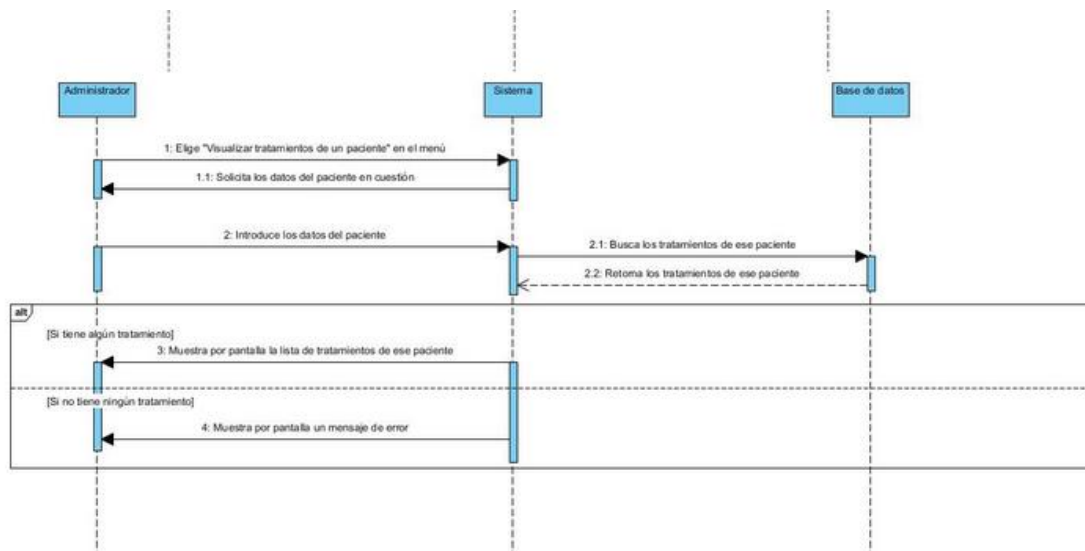


Figura 6.12: Visualizar tratamientos

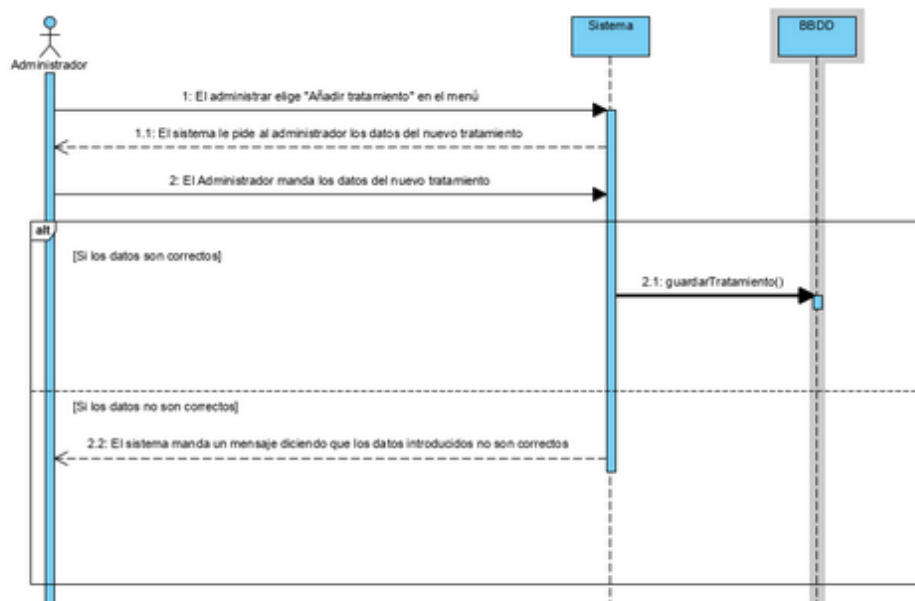


Figura 6.13: Añadir tratamiento

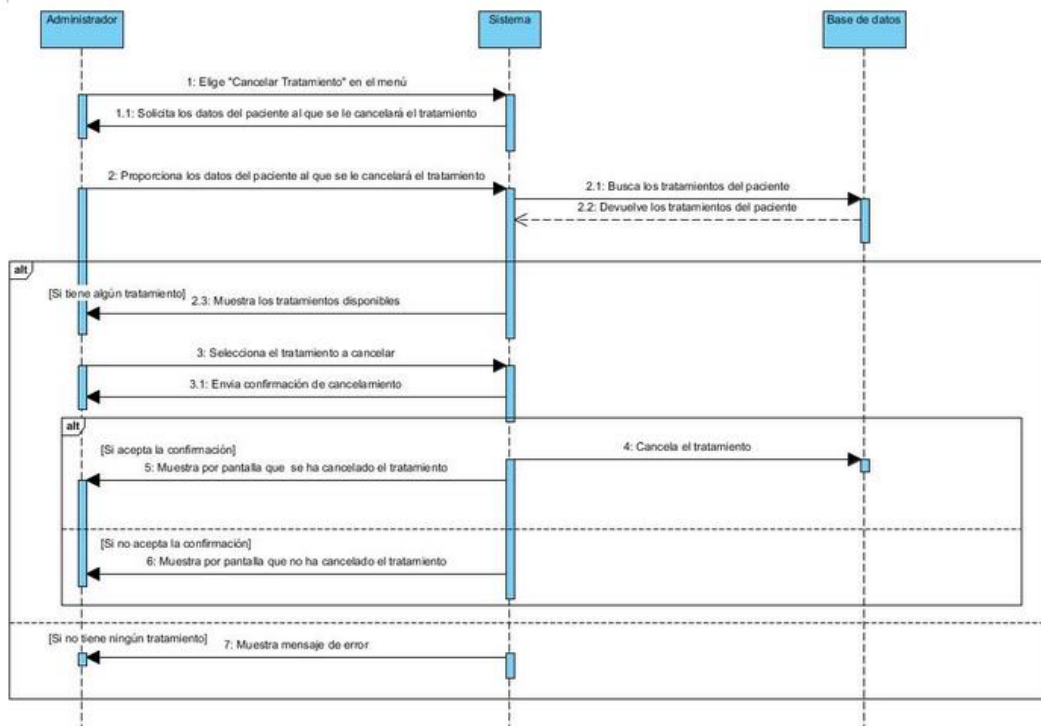


Figura 6.14: Cancelar tratamiento

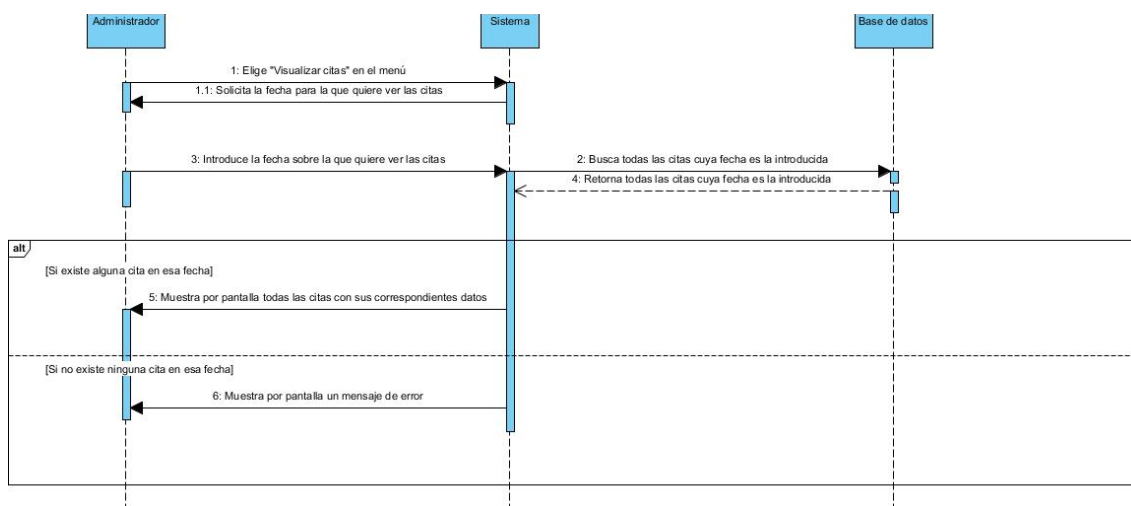


Figura 6.15: Visualizar citas día actual

7. Matrices validación

	CU1	CU2	CU3	CU4	CU5	CU6	CU7	CU8	CU9	CU10	CU11	CU12	CU13	CU14	CU15
RF1	X														
RF2		X		X	X	X	X	X	X	X	X	X	X	X	
RF3			X												
RF4				X											
RF5					X										
RF6						X									
RF7							X								
RF8								X							
RF9									X						
RF10										X					
RF11											X				
RF12												X			
RF13													X		
RF14														X	
RF15															X

	Paciente	Cita	Historial	Tratamiento
CU1	X			
CU2	X			
CU3	X			
CU4	X			
CU5	X			
CU6	X	X		
CU7	X	X		
CU8	X	X		
CU9	X	X		
CU10	X		X	
CU11	X		X	
CU12	X			X
CU13	X			X
CU14	X			X
CU15		X		

8. Procedimiento scrum

Scrum es una metodología ágil y flexible para gestionar el desarrollo de software de manera continua, es decir, nos permite gestionar el tiempo que se va a utilizar para realizar las tareas deseadas.

Para el desarrollo de esta metodología utilizaremos un listado con las tareas a realizar indicando el miembro del grupo de trabajo que lo va a realizar y el tiempo que tiene para terminar dicha tarea.

Este listado de tareas en nuestro caso se dividirá en 4 sprint. Cada tarea tendrá asignada una serie de puntos de tiempo, que pretenderá ofrecer una estimación de la duración de la implementación de cada tarea. Cada sprint durará una semana.

Sprint 1: 18 Noviembre 2019 - 24 Noviembre 2019

Sprint 2: 25 Noviembre 2019 – 1 Diciembre 2019

Sprint 3: 2 Diciembre 2019 – 8 Diciembre 2019

Sprint 4: 9 Diciembre 2019 – 15 Diciembre 2019

TAREA	SPRINT	PUNTOS TIEMPO	RESPONSABLE
Añadir cita	1	3	Adrián
Modificar cita	1	4	Gabriel
Cancelar cita	1	2	Javier
Dar de alta nuevo paciente	2	2	Adrián
Buscar paciente	2	2	Alicia
Modificar datos paciente	2	4	Javier
Eliminar paciente	2	2	Gabriel
Consultar cita de un paciente	3	3	Gabriel
Cancelar tratamiento	3	3	Javier
Generar lista pacientes	3	1	Alicia
Añadir tratamiento	3	4	Adrián
Importar fichero macros	3	2	Alicia
Añadir información historial clínico	4	4	Adrián
Visualizar citas día actual	4	4	Alicia
Visualizar tratamientos	4	2	Gabriel
Ver historial clínico	4	2	Javier