
GESTOR DE TAREAS ROBOTS ASISTENCIALES

grupo rojo

Ingeniería Web

Versión 2.3

Realizado por: Alejandro Domínguez Recio
Hugo Ávalos de Rorthais
Rosario García Morales
Laura Núñez Jiménez
Jesús Aldana Martín

Profesor : Javier Cámara Moreno

15 de diciembre de 2022

Índice general

1	Introducción	4
1.1	Objetivos	4
1.2	Formato utilizado	4
2	Descripción de producto	6
2.1	Características de los usuarios y entorno	6
2.1.1	Resumen de los usuarios	6
2.1.2	Perfiles de usuarios	7
2.2	Funcionalidad del producto	7
3	Requisitos funcionales	8
3.1	Requisitos funcionales	8
3.1.1	FR 0	8
3.1.2	FR 1	8
3.1.3	FR 2	9
3.1.4	FR 3	9
3.1.5	FR 4	9
3.1.6	FR 5	10
3.1.7	FR 6	10
3.1.8	FR 7	11
3.1.9	FR 8	11
3.1.10	FR 9	11
3.1.11	FR 10	12
3.1.12	FR 11	12
4	Requisitos no funcionales	13
4.1	Requisitos de almacenamiento	13
4.1.1	NFR 0	13
4.1.2	NFR 1	13
4.2	Requisitos de seguridad	14
4.2.1	NFR 2	14
4.2.2	NFR 3	14
4.2.3	NFR 4	15
4.3	Requisitos de diseño	15
4.3.1	NFR 5	15
4.3.2	NFR 6	16
4.3.3	NFR 7	16

5	Diseño	17
5.1	Modelado de procesos	18
5.1.1	Técnico	18
5.1.2	Médico	18
5.2	Diagramas IFML	19
5.2.1	Técnico	20
5.2.2	Médico	21
5.3	Modelado de Contenidos	22
5.3.1	Descripción de clases	23

1 Introducción

La sanidad pública en muchos países se ve saturada por la alta demanda de pacientes enfermos o que requieren de un diagnóstico, esto provoca grandes retrasos a la hora de pedir citas y aumenta la probabilidad de que produzcan problemas humanos por el nivel de saturación que encontramos en algunos hospitales. Es por ello que los robots asistenciales están tomando mucho protagonismo en el ámbito sanitario. Estos pueden mejorar el rendimiento y optimizar trabajos para ayudar así al sanitario. Para mejorar esta situación se han comenzado a incorporar en los hospitales robots controlados por una aplicación que ayuden a tener una mejor comunicación interna dentro del hospital y liberar a los sanitarios, enfermeros y auxiliares de tareas rutinarias que a día de hoy pueden resolver robots sin ningún tipo de problema.

1.1. Objetivos

Durante el desarrollo de nuestra aplicación buscamos resolver los siguientes escenarios: Con el desarrollo de este proyecto perseguimos crear una aplicación web que comunique de forma remota al sanitario con los robots del hospital, principalmente nos centraremos en la inserción y modificación de tareas desde la web. Para que el robot se desplace automáticamente tras recibir alguna petición por parte del sanitario. Los robots y, por tanto, nuestra app se centrara en tareas especializadas en desinfección, transporte de medicamentos y teleasistencia.

- Permitir la comunicación entre el sanitario y el robot.
- Una buena gestión de manejo de tareas de los robots asistenciales.
- Agilizar el trabajo del personal sanitario.

1.2. Formato utilizado

Se ha decidido usar una tabla por requisito. En cada tabla se añadirá la descripción del requisito, la prioridad que tiene respecto a los demás, el riesgo de no cumplirse y una descripción del mismo, el motivo del requisito. Se implementa también una parte de criterios de aceptación para saber que el requisito ha sido implementado adecuadamente. Se utiliza la misma plantilla para FR y NFR, porque son requisitos como tal, y su estructura son parecidas. El motivo por el cual se ha decidido usar esta implementación es la sencillez de visualización tanto para los realizadores del proyecto como para el cliente a la hora de mostrarle estos requisitos, esto es más sencillo, debido a que los

requisitos van a tener una estructura organizada y esquematizada. Cuando lo veamos ya sabemos directamente muchas funcionalidades del requisito.

2 Descripción de producto

El producto trata de una aplicación web orientada a la gestión de robots de un centro sanitario. Esta aplicación por parte del sanitario se usará para gestionar los robots y obtener 'feedback' sobre su estado. Por parte del técnico, modificará la plantilla de robots, definirá los posibles comportamientos y tratará con los errores de los mismos.

2.1. Características de los usuarios y entorno

Definiendo como usuario a toda aquella persona que interactúa directamente con la aplicación. En esta sección se detallaran las principales características de los usuarios así como del uso que harán estos de la aplicación.

2.1.1. Resumen de los usuarios

Nombre	Descripción	Relaciones
Técnico	Encargado de la gestión técnica de la flota de robots. Realiza las tareas de incorporación o retirada de robots, configuración de tareas o apoya en caso de incidencias	Relación Técnico-Sanitario
Sanitario	Encargado de la gestión del estado de la flota de robots. Realiza las tareas de asignación, modificación o eliminación de las tareas asignadas a los robots.	Relación Sanitario-Técnico

Cuadro 2.1: Resumen usuarios

2.1.2. Perfiles de usuarios

TÉCNICO	
Representante	Técnico.
Descripción	Persona con conocimientos técnicos sobre los robots. Aplica los conocimientos técnicos en la configuración, mantenimiento, reparación y solución de problemas de estos.
Responsabilidades	<ul style="list-style-type: none">• Incorporar o retirar robots de la flota• Definir el conjunto de tareas permitidas en los robots (Dentro de su tipo).• Modificar características técnicas de los robots (Ej : Modo energía)• Asistir incidencias técnicas
Criterio de éxito	Proporcionar un sistema que cubra sus responsabilidades de una forma simple.

Cuadro 2.2: Tabla Usuario_1

SANITARIO	
Representante	Sanitario.
Descripción	Persona con conocimientos sobre tratamientos o cuidados aplicados a la salud de las personas. Identifica, realiza y requiere de actividades aplicadas en su práctica clínica.
Responsabilidades	<ul style="list-style-type: none">• Asignar tareas a los robots.• Fijar prioridad a las tareas.• Fijar tiempo de realización a las tareas• Modificar estado de tareas.• Revisar estado de los robots y tareas asociadas
Criterio de éxito	Proporcionar un sistema que cubra sus responsabilidades de una forma simple.

Cuadro 2.3: Tabla Usuario_2

2.2. Funcionalidad del producto

El producto mejorará la organización de tareas de un conjunto de robots. Este implementará una app donde tenga una interfaz orientativa para el médico y así se obtenga una buena comunicación con los robots.

3 Requisitos funcionales

A continuación los requisitos funcionales desarrollados para la aplicación. Cada tabla corresponde a un requisito diferente cada requisito es nombrado por un identificador único, en nuestro caso se trata de "FR_NúmeroRequisito" cada fila de la tabla revela información adicional sobre el requisito en cuestión.

3.1. Requisitos funcionales

3.1.1. FR 0

FR_0	
Descripción	El técnico podrá incorporar y eliminar usuarios de la aplicación
Prioridad	Alta
Riesgo	Nulo
Descripción del riesgo	Nulo
Razón	Proporcionar funcionalidad de gestión básica al técnico
Criterio de aceptación	El usuario nuevo aparece en la base de datos

Cuadro 3.1: Requisito Funcional FR_0

3.1.2. FR 1

FR_1	
Descripción	El técnico podrá definir el conjunto de tareas permitidas en los robots
Prioridad	Alta
Riesgo	Nulo
Descripción del riesgo	Nulo
Razón	Proporcionar funcionalidad de gestión básica al técnico
Criterio de aceptación	La tarea aparecerá en la vista de lista de tareas

Cuadro 3.2: Requisito Funcional FR_1

3.1.3. FR 2

FR_2	
Descripción	El técnico puede ver la asistencias de problemas técnicos.
Prioridad	Alta
Riesgo	Nulo
Descripción del riesgo	Nulo
Razón	Proporcionar funcionalidad de atención técnica a los sanitarios.
Criterio de aceptación	Las incidencias aparecen en la ventana de incidencias

Cuadro 3.3: Requisito Funcional RF_2

3.1.4. FR 3

FR_3	
Descripción	El sistema debe posibilitar al técnico la monitorización de robots
Prioridad	Alta
Riesgo	Nulo
Descripción del riesgo	Nulo
Razón	Proporcionar funcionalidad de gestión básica al técnico. La monitorización de los distintos robots es clave a la hora planificar o asignar tareas.
Criterio de aceptación	

Cuadro 3.4: Requisito Funcional RF_3

3.1.5. FR 4

FR_4	
Descripción	El sanitario dispondrá de una herramienta de filtrado para los robots.
Prioridad	Media
Riesgo	Medio
Descripción del riesgo	Problemas de implementación. Incremento de complejidad del sistema. Situaciones de uso no tenidas en cuenta.
Razón	Facilitar la navegación del sanitario
Criterio de aceptación	El filtrado será correspondiente a cada estado del robot.

Cuadro 3.5: Requisito Funcional RF_4

3.1.6. FR 5

FR_5	
Descripción	El sanitario podrá asignar tareas.
Prioridad	Máxima
Riesgo	Máximo
Descripción del riesgo	Toda la aplicación gira entorno a la asignación de tareas del sanitario al robot.
Razón	Al guardar las tareas en una cola tendremos organizadas las tareas a realizar.
Criterio de aceptación	Asignación de tarea correcta y aparezca en la lista de tareas del robot.

Cuadro 3.6: Requisito Funcional RF_5

3.1.7. FR 6

FR_6	
Descripción	El sanitario tendrá la posibilidad de dar diferentes niveles de prioridad.
Prioridad	Máxima
Riesgo	Medio
Descripción del riesgo	Problemas de implementación algorítmica
Razón	Los robots tienen que hacer las tareas urgentes antes que las demás. A más alta automáticamente se colocarán por encima de otras más triviales en la cola.
Criterio de aceptación	Asignación de tarea correcta y aparezca en la lista de tareas del robot como la primera.

Cuadro 3.7: Requisito Funcional RF_6

3.1.8. FR 7

FR_7	
Descripción	Las tareas de prioridad 'Máxima' interrumpirán la tarea que esté desempeñando el robot para el cumplimiento de la misma.
Prioridad	Alta
Riesgo	Nulo
Descripción del riesgo	Nulo
Razón	Proporcionar funcionalidad de gestión básica del sistema.
Criterio de aceptación	La tarea de urgencia maxima aparece la primera en el listado de tareas.

Cuadro 3.8: Requisito Funcional RF_7

3.1.9. FR 8

FR_8	
Descripción	Las tareas interrumpidas se reasignan/eliminan de la cola según su prioridad
Prioridad	Alta
Riesgo	Medio
Descripción del riesgo	Problema de implementación logarítmica. Que no aparezcan en la lista o que sigan apareciendo. Haría que entrará en bucle la lista de tareas.
Razón	Proporcionar funcionalidad de gestión básica del sistema.
Criterio de aceptación	Las tareas interrumpidas se hayan eliminado del historial del robot.

Cuadro 3.9: Requisito Funcional RF_8

3.1.10. FR 9

FR_9	
Descripción	La aplicación debe tener registro e inicio de sesión.
Prioridad	Alta
Riesgo	Alto
Descripción del riesgo	No saber a que funcionalidad dar acceso.
Razón	Debemos conocer en todo momento que usuarios acceden a la app.
Criterio de aceptación	El usuario entrará en su interfaz correspondiente.

Cuadro 3.10: Requisito Funcional RF_9

3.1.11. FR 10

FR_10	
Descripción	Toda los datos que genere la app serán recogidos y almacenados.
Prioridad	Alta
Riesgo	Medio
Descripción del riesgo	Perder información que puede ser valiosa o necesaria para la app.
Razón	Por motivos de seguridad guardaremos la información de quien ha tenido acceso a la app.
Criterio de aceptación	Asignación de tarea correcta y aparezca en la lista de tareas del robot.

Cuadro 3.11: Requisito Funcional RF_10

3.1.12. FR 11

FR_11	
Descripción	Debe haber un registro de todas las tareas hecha por cada robot
Prioridad	Alta
Riesgo	Bajo
Descripción del riesgo	Perder información que puede ser valiosa o necesaria para la app.
Razón	Por motivos de seguridad guardaremos la información de que ha hecho cada robot, en que momento y quien se lo ha ordenado.
Criterio de aceptación	En el listado de tareas tanto para el sanitario como para el técnico aparezcan todas las tareas propuestas.

Cuadro 3.12: Requisito Funcional RF_11

4 Requisitos no funcionales

4.1. Requisitos de almacenamiento

4.1.1. NFR 0

NFR_0	
Descripción	Se utilizarán herramientas de almacenamiento en la nube para guardar los datos de la app.
Prioridad	Media
Riesgo	Medio
Descripción del riesgo	Perder información valiosa.
Razón	Los datos en la nube están más seguros que en local.
Padres	FR_10
Criterio de aceptación	Todos los datos aparecieran en la nube

Cuadro 4.1: Requisito No Funcional NRF_0

4.1.2. NFR 1

NFR_1	
Descripción	El tiempo de carga de página web deberá de ser menor a 2s
Prioridad	Alta
Riesgo	Bajo
Descripción del riesgo	Incompatibilidad con características de diseño
Razón	El tiempo de carga es fundamental en la experiencia de usuario
Padres	NFR_2
Criterio de aceptación	El tiempo estimado fue de 2 segundos o menos.

Cuadro 4.2: Requisito No Funcional NRF_1

4.2. Requisitos de seguridad

4.2.1. NFR 2

NFR_2	
Descripción	A la hora del registro, el usuario debe especificar si es sanitario o técnico
Prioridad	Alta
Riesgo	Alto
Descripción del riesgo	De no ser así, no podríamos diferenciar entre clínico o técnico.
Razón	Ambos tipos de usuario podrían acceder a las funciones del otro.
Padres	FR_9
Criterio de aceptación	El usuario si tiene acceso correspondiente a su cargo le saldrá la interfaz adecuada

Cuadro 4.3: Requisito No Funcional NRF_2

4.2.2. NFR 3

NFR_3	
Descripción	Tras un registro la contraseña sera encriptada para guardarla como un id único y no tener acceso a ella.
Prioridad	Baja
Riesgo	Medio
Descripción del riesgo	Acceder a los datos privados de nuestros usuarios.
Razón	Para no tener en ningún momento acceso a las contraseñas de los usuarios.
Padres	FR_9
Criterio de aceptación	Aviso de contraseña encriptada

Cuadro 4.4: Requisito No Funcional RNF_3

4.2.3. NFR 4

NFR_4	
Descripción	El sistema deberá de mantener un registro de usuario/clave única.
Prioridad	Alta
Riesgo	Bajo
Descripción del riesgo	Funcionalidad de clave única proporcionada por la base de datos implementada
Razón	Controlar el acceso a cuentas, como la identidad única de los usuarios.
Padres	FR_9
Criterio de aceptación	Si el técnico añade un usuario le saldrá un aviso de que el usuario no existe en la base de datos, y podrá utilizar ese ID.

Cuadro 4.5: Requisito No Funcional NFR_4

4.3. Requisitos de diseño

4.3.1. NFR 5

NFR_5	
Descripción	La web debe tener un diseño accesible y simple.
Prioridad	Alta
Riesgo	Alto
Descripción del riesgo	Que un sanitario sin ningún background tecnológico no sepa como trabajar con la app.
Razón	La app debe de ser auto explicativa para que cualquier usuario sepa utilizarla.
Padres	FR_10
Criterio de aceptación	La buena facilidad de manjera la aplicación.

Cuadro 4.6: Requisito No Funcional NRF_5

4.3.2. NFR 6

NFR_6	
Descripción	Según el tipo de usuario, la app tendrá una u otra interfaz.
Prioridad	Alta
Riesgo	Medio
Descripción del riesgo	De no ser así, no podríamos diferenciar entre clínico o técnico.
Razón	Ambos tipos de usuario podrían acceder a las funciones del otro.
Padres	FR_9
Criterio de aceptación	La interfaz correspondiente a cada usuario.

Cuadro 4.7: Requisito No Funcional NFR_6

4.3.3. NFR 7

NFR_7	
Descripción	El sistema debe posibilitar la creación de una tarea con X clicks desde la vista monitorización del usuario
Prioridad	Alta
Riesgo	Baja
Descripción del riesgo	Dificultad de crear lógica de procesos.
Razón	La creación de tareas de una forma 'rápida' es fundamental en la experiencia de usuario. Se tiene en cuenta que la creación de tareas es una función principal de la aplicación
Criterio de aceptación	Asignación de tarea correcta y fácil.

Cuadro 4.8: Requisito No Funcional NFR_7

5 Diseño

El modelado web es una rama de la ingeniería web que aborda los problemas específicos relacionado con el diseño y desarrollo. En particular, se centra en el proceso de documentar un diseño de un sistema software complejo en un diagrama de flujo de estados de fácil comprensión utilizando texto y símbolos para representar la forma en la que el usuario accederá a la aplicación. Encontramos varios enfoques para el modelado web:

- Modelo de procesos: Es una expresión abstracta de los procesos principales de un sistema se representa de manera secuencial.
- Diagramas IFML: Representa los elementos de la interfaz gráfica de la aplicación así como la forma en que el usuario pasa de un elemento a otro.
- Modelo de contenidos: Proporcionan una visión orientada a la ejecución, dan al desarrollador información a nivel de sistema.

Para tener una idea de como íbamos a realizar los diferentes modelos utilizamos una herramienta de dibujo para crear un boceto en el que mostramos de forma simple y mediante símbolos lo que contendrán en un futuro las diferentes vistas.

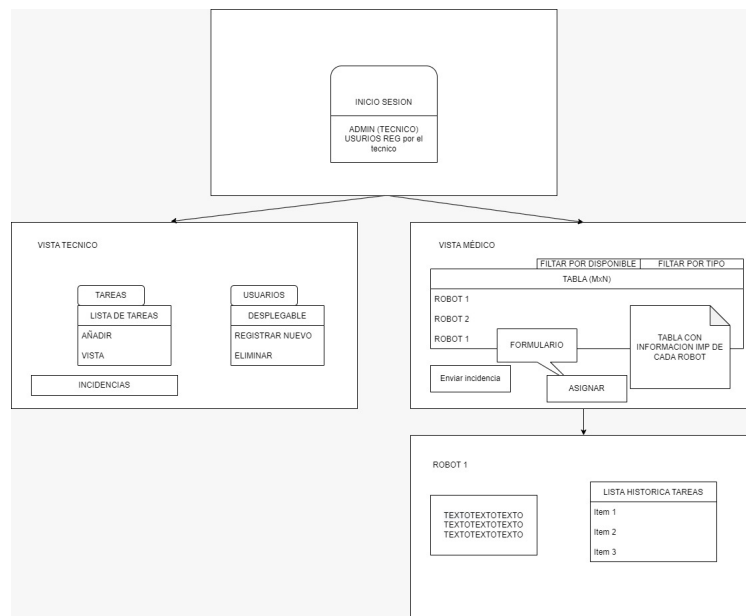


Figura 5.1: Boceto inicial

5.1. Modelado de procesos

El modelado de procesos es una expresión abstracta de los principales procesos de un sistema, y se suelen representar de manera secuencial. En los procesos se describe en que orden un grupo de objetos funcionan entre sí son utilizados para comprender mejor los requisitos de un sistema nuevo que se implemente o un proceso existente ya que nos permite visualizar de forma gráfica como se realizaran diferentes funcionalidades de la web.

Dado que nuestra app presenta dos tipos de usuarios muy diferenciados hemos creado un proceso para cada uno de ellos en los que se muestra todas las opciones que tiene disponible y sus funcionalidades.

5.1.1. Técnico

Cuando el usuario entra a la app web, el usuario se encuentra un login procede a poner usuario y contraseña, se confirma que el usuario está en la base de datos y entra a la interfaz correspondiente. El usuario puede volver al login o ir a lista de usuarios o tareas, donde desde el botón de usuario puede eliminar o añadir, y en la lista puede añadir y ver. Una vez realizada las funciones, se actualiza en la base de datos se confirma y se vuelve a la interfaz del técnico.

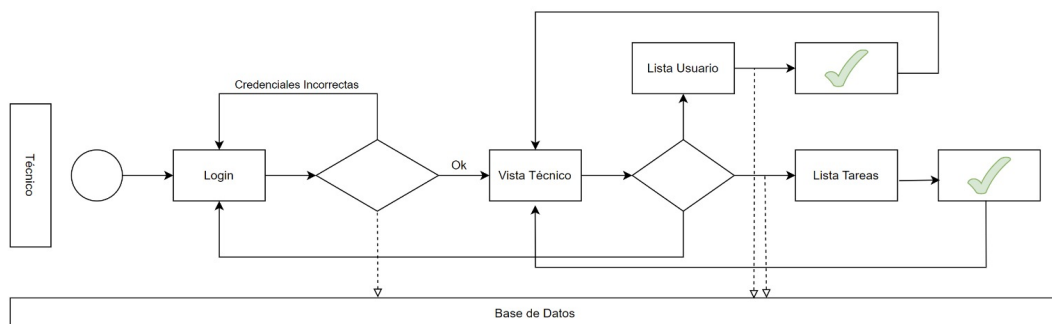


Figura 5.2: Diagrama de procesos del técnico

5.1.2. Médico

Al igual que para el técnico, el médico al acceder a la web tiene un inicio de sesión, donde deberá introducir sus credenciales que serán comprobadas en la base de datos. En caso de éxito podrá acceder a la interfaz y sino seguirá en la pestaña de login. Una

vez haya accedido a la interfaz de la app, podrá elegir entre diferentes opciones, asignar tareas, notificar una incidencia sobre algún robot, ver la lista de robots o cerrar sesión si así lo desea. En caso de consultar la lista de robots, podrá acceder a una nueva vista con toda la información el robot elegido. En todo momento tras cada vista se harán consultas sobre la base de datos para mostrar información por pantalla o se guardará en la base de datos las incidencias o tareas asignadas.

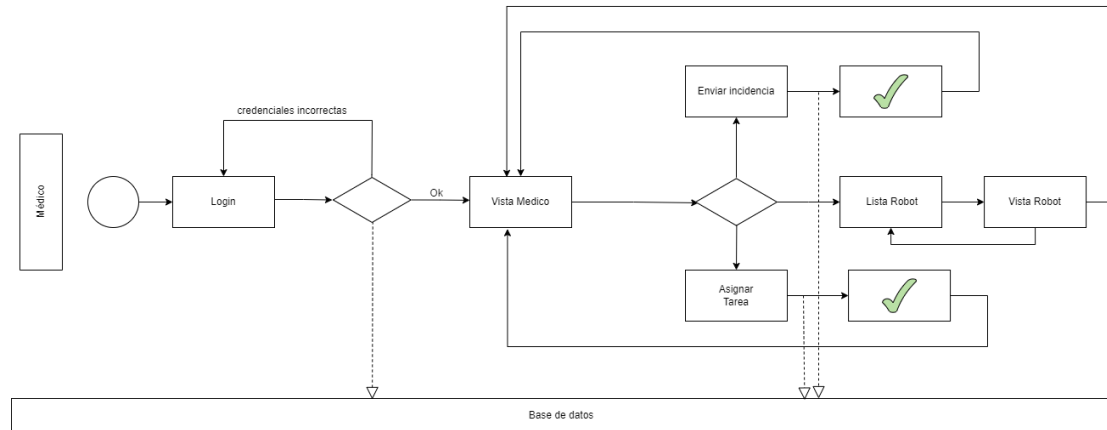


Figura 5.3: Diagrama de procesos del médico

5.2. Diagramas IFML

El lenguaje de **modelado de flujo de interacción** (IFML) incluye un conjunto de notaciones gráficas para crear modelos visuales de interacciones de usuario y comportamiento de front-end en sistemas de software.

Los modelos IFML constan de uno o más contenedores de vista. Estos pueden incluso abarcar otro contenedor de ser necesario, las vistas denotan el contenido estático o dinámico de la página así como los elementos de interfaz para la entrada de datos. Tanto los contenedores como los componentes de una vista se pueden asociar entre si con eventos. Los eventos son situaciones que afectan al estado de la página, la forma de relacionar una vista con una acción es mediante los flujos de navegación que son flechas que nos indican el sentido del flujo.

Algunos beneficios que aporta el desarrollo en IFML:

- Permite abordar el modelado desde diferentes perspectivas como son: interfaz, interacción y gestión de eventos.
- Aísla los problemas de implementación de front-end con el diseño permitiendo obtener un concepto general del mismo sin preocupaciones.
- Mejora la comunicación del diseño de la interfaz de usuario con las partes interesadas no técnicas.

5.2.1. Técnico

Después de iniciar sesión con la cuenta del técnico accedemos a su interfaz. Desde esta vista podemos elegir entre tres acciones, cada una representada por un botón diferente. La primera opción sería el botón "Incidencias", el cual nos lleva a una nueva vista donde nos saldrá si hay alguna incidencia con algún robot y sus detalles. El siguiente botón que tenemos en la vista principal es "Usuario", este también nos redirige a otra vista donde nos saldrá una lista con los usuarios registrados en el sistema. Desde esta lista podremos eliminar algún usuario existente o registrar algunos nuevos. Para ello usaríamos el botón "UsuarioNuevo" se nos abriría un formulario donde se introducen los datos para iniciar sesión del nuevo usuario. La última funcionalidad es "Lista de tareas", este botón nos carga una nueva vista donde veremos una lista con todas las tareas definidas que pueden realizar los robots. Desde esta vista podemos añadir una tarea mediante el formulario que se despliega o pinchar sobre una ya existente, lo que nos llevaría a otra ventana con la tarea elegida y su información.

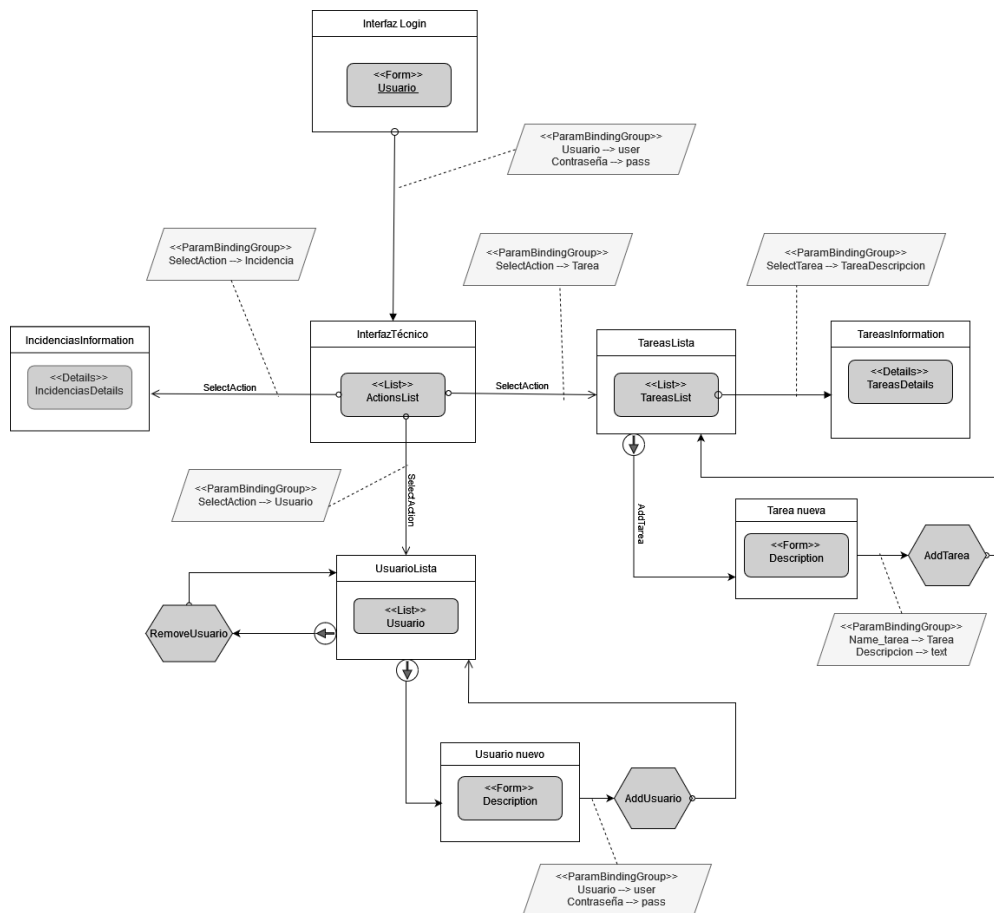


Figura 5.4: Diagrama IFML médico

5.2.2. Médico

Tras hacer login con el usuario y la contraseña del médico accederemos directamente a su interfaz correspondiente. Está es la vista general del médico y en ella aparecen diversos botones, cada botón realiza una función diferente, esto lo podemos observar en el diagrama IFML, dentro del contenedor de interfaz encontramos una «List» *ActionList*. Del contenedor salen diferentes flujos de navegación. Leyendo el diagrama de izquierda a derecha la primera funcionalidad de la interfaz de médico que vemos es la de las incidencias. Accedemos a está tras pulsar el botón de enviar incidencias, la vista de las incidencias posee un textbox en el cual el usuario podrá escribir cualquier mensaje que quiera que vea el técnico es por eso que debajo del textbox aparecen dos botones de evento, uno envía el mensaje a la vista del técnico y el otro cancela el proceso devolviéndonos a la interfaz anterior. La segunda funcionalidad es la de asignación de tareas mediante un formulario que aparecerá por pantalla el médico podrá asignar tareas tras asignar o cancelar una tarea volveremos a la interfaz anterior. Por último tenemos la lista de robots donde podremos pinchar en cualquiera de ellos para ver una descripción general de las características del robot.

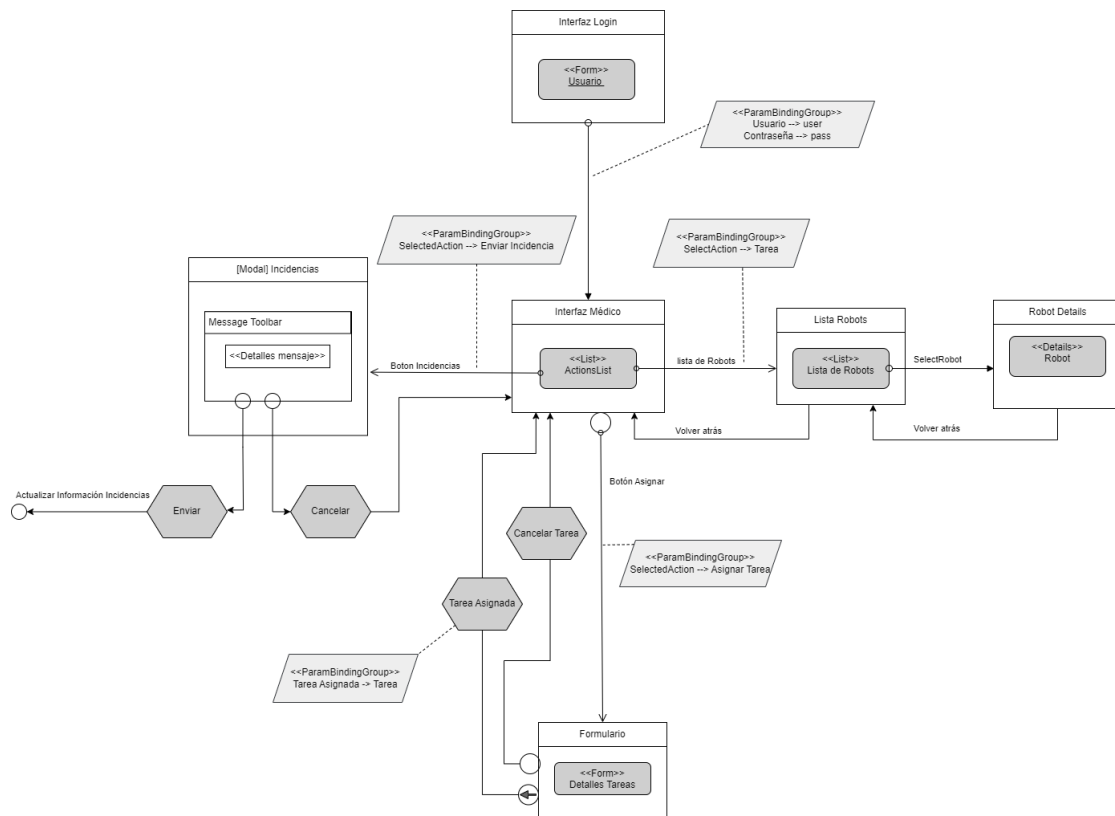


Figura 5.5: Diagrama IFML técnico

5.3. Modelado de Contenidos

En el modelado de contenidos se representa la estructura, el comportamiento y relaciones estáticas de cada uno de los objetos del sistema.

El modelo de contenidos esta formado principalmente por los siguientes elementos: **clases, relaciones e interfaces.**

- Las clases representan entidades u objetos del sistema y tienen atributos y métodos asociados que las definen.
- Las relaciones representan como se relacionan los objetos del sistema. Las propiedades definen las relaciones son la multiplicidad(número de elementos que participan en la relación) y el nombre de asociación.
- Las interfaces son definidas por una serie de atributos, funciones y obligaciones. Toda clase asociada a una interfaz debe de cumplir con todas aquellas funciones y atributos incluidos en esta.

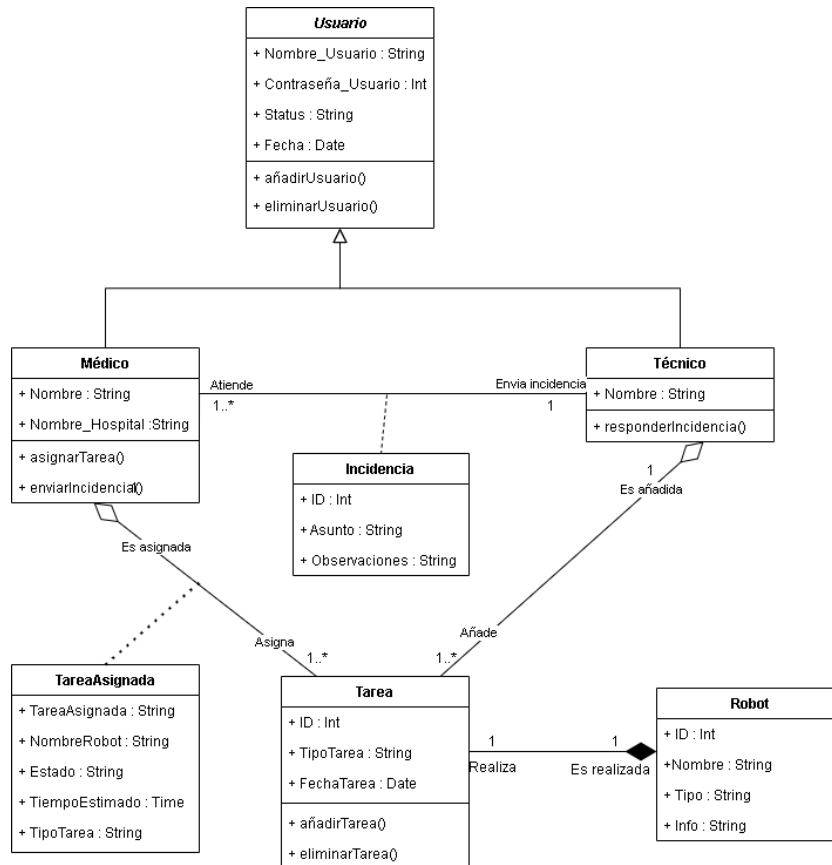


Figura 5.6: Modelo de contenidos

5.3.1. Descripción de clases

- **Clase Usuario:** Aquellas personas que van hacer uso de la web. Pueden ser médicos o técnicos.
- **Clase Médico:** Un tipo de usuario. Es el encargado de asignar tareas a los robots y en el caso de incidencias con las plataforma, comunicárselas al técnico.
- **Clase Técnico:** Un tipo de usuario. Es el encargado de añadir tareas a los robots y atender las incidencias comunicadas por los médicos.
- **Clase Incidencia:** Modela las incidencias registradas por lo médicos. Dicho objeto almacenas las observaciones realizadas por lo médicos y técnicos.
- **Clase Tarea:** Modela las tareas asignadas y añadidas por los médicos y técnicos respectivamente. Una tarea solo puede ser asignada por un médico. A la vez una tarea solo puede ser añadida por un técnico. Todas las tareas tendrán asignadas un robot que la realice. Una tarea no puede existir sin un técnico que previamente la halla añadido y un robot asociado.

- **Clase Robot:** Modela los robots de la plataforma. Los robots pueden realizar un tipo de tarea.