

Documentando la Arquitectura



Integrantes:

Lucas Fernández Liguori, bo0405

Rafael Sánchez Rodríguez, bm0323

Jose Carlos Gago Hernández, bm0031

María Fernández Herraiz, bm0553

Jonatan González Jiménez bj0563

ÍNDICE

1.- Introducción: descripción del problema

1.1.- Descripción general del problema

La tecnología sanitaria ha evolucionado mucho en los últimos tiempos, llegando a conseguir maquinaria que puede realizar operaciones complejas manejadas por un profesional a miles de kilómetros de distancia. La adquisición de éste tipo de máquinas por parte de los hospitales, ha generado que sea prioritario alguna manera de gestionar la correcta utilización de las mismas, así como su control y conocimiento de los operarios capacitados para controlarlas.

1.2.- Business Goals

El propósito del proyecto propuesto es reducir el tiempo que se utilizarán los recursos de los que disponen las infraestructuras de sanidad, lo que a su vez se verá traducido en una mayor capacidad de atención a los enfermos y una reducción de las esperas.

- BG-0: Para el sistema Sforza que está siendo desarrollado, el ministro de sanidad desea que se construya un sistema de gestión de robots Da Vinci para su uso en el sistema sanitario que dure 10 años sin necesidad de cambiarlo.
- BG-1: Para el sistema Sforza que está siendo desarrollado, el ministro de sanidad desea que los datos guardados en la base de datos cumplan con los principios de integridad, confidencialidad y disponibilidad y será satisfecha si aprueban el estándar ISO 27001.
- BG-2: Para el sistema Sforza que está siendo desarrollado, el ministro de sanidad desea que el sistema esté siempre operativo a pleno funcionamiento y será satisfecho si el tiempo de mantenimiento es inferior a 2h.
- BG-3: Para el sistema Sforza que está siendo desarrollado, el ministro de sanidad desea que se conozca con datos precisos el estado de los dispositivos Da Vinci para su correcto funcionamiento y será satisfecha si se pueden monitorizar los parámetros de todos los robot.

- BG-4: Para el sistema Sforza que está siendo desarrollado, el personal técnico hospitalario desea que el sistema responda a cualquier petición que se le haga de forma rápida, eficaz y sin bloquearse y será satisfecha si el tiempo de respuesta a cualquier petición al sistema se completa en menos de 1 segundo.
- BG-5: Para el sistema Sforza que está siendo desarrollado, el personal hospitalario desea que el sistema sea fácil de usar y de aprender a usar y será satisfecha si el tiempo de aprendizaje es menor a 2.5 horas.
- BG-6: Para el sistema Sforza que está siendo desarrollado, la administración de los hospitales desea que el sistema sea ampliable tanto en número de robots como en funcionalidad y será satisfecha si dicha ampliación se puede realizar en menos de dos semanas.

1.3.- Business Drivers

El grupo propone el diseño de un sistema que gestione de forma óptima la utilización de los robots, su mantenimiento y los suministros hardware que se necesiten para su correcto funcionamiento.

2.- Stakeholders

2.1.- Lista justificada y priorizada de stakeholders

1. Ministro de Sanidad
2. Administración del hospital
3. Gerente/s hospital.
4. Personal médico.
5. Personal de enfermería.
6. Personal técnico:
 - 6.1 Personal de informática.
 - 6.2 Personal de prevención de riesgos laborales.
7. Arquitecto de sistemas
8. El equipo de desarrollo

El Ministro de Sanidad es quién encarga el proyecto y el product owner, da las directrices principales necesarias expresadas en los atributos de calidad. En un siguiente nivel está el personal del hospital que se va a encargar de manejar el sistema, dividido en varios grupos: la administración, el gerente y el personal médico y de enfermería. Siguiendo el personal que se encargará del constante funcionamiento del sistema y de su mantenimiento, véanse el personal de informática y el de prevención de riesgos laborales. También hicimos uso de un

arquitecto de sistemas experto en sistemas para sanidad como asesor. Por último el propio equipo de desarrollo.

Tuvimos una serie de sesiones de entrevistas con el ministerio de sanidad, donde pudimos obtener en mayor medida los atributos de calidad necesarios para la construcción del sistema, que luego fueron refinados en el siguiente apartado, después de dichos eventos con algunos Stakeholders y el Analista de Negocio.

3.- Atributos de calidad (QA)

3.1.- Descripción de los atributos de calidad más importantes y su priorización justificada

A la hora de priorizar los atributos, hemos decidido separarlos en varios grupos, teniendo en cuenta su importancia:

En este primer grupo, tenemos los atributos que mantienen al sistema funcionando de manera correcta y de forma ininterrumpida.

- 1.-Servicio continuo.
- 2.-Segura (Privacidad, integridad, confidencialidad).
- 3.-Control de estado del robot.
- 4.-Prevención de fallos.
- 5.-Prevención de caídas de sistema.
- 6.-Carga del sistema.
- 7.-Tiempo de proceso.

En el siguiente bloque tenemos los atributos que permiten al sistema tener control sobre los datos y hardware.

- 8.-Comunicación con los robots(Interfaces externa, interoperabilidad).
- 9.-Safety.
- 10.-Datos correctos.
- 11.-Tiempo de respuesta de BD.
- 12.-Comunicación entre sistemas.
- 13.-Fácilmente modificable.
- 14.-Optimizado.

Por último, los atributos orientados al usuario y que no influyen en el funcionamiento del sistema.

- 15.-Fácil de usar.
- 16.-Fácil de aprender.
- 17.-Fácil de mantener.
- 18.-Fácil de administrar.
- 19.-Fácil de testear.
- 20.-Fácil de portar.

- 21.-Fácil de verificar.
- 22.-Reciclaje de componentes.
- 23.-Capacidad de expandir la funcionalidad del sistema.

3.2.- Árbol de utilidad

Atributo de calidad	Atributo refinado	ASR
Disponibilidad	Servicio continuo.	DIS-1. El sistema deberá estar disponible las 24h del día, 7 días a la semana, exceptuando mantenimientos del mismo, los cuales deberán ser entre las 01:00AM y las 03:00AM.
Security	Privacidad	SEC-1. El sistema mantendrá todos los datos personales y confidenciales invisibles a los usuarios y agentes externos al personal del mismo y a aquellos que no tenga los permisos necesarios para visualizarlos.
	Integridad	SEC-2. El sistema protegerá los datos ante modificaciones no autorizadas, al igual que será capaz de verificar las copias de seguridad de los ficheros, informando de cualquier tipo de discrepancia.
	Confidencialidad	SEC-3. El sistema evitará que se impriman todos los datos personales y confidenciales.
Precisión	Control de estado del robot	PRE - 1. El usuario podrá comprobar el estado del robot con datos precisos.
	Datos correctos	PRE - 2. Los datos proporcionados al usuario serán correctos.
Fiabilidad	Prevención de fallos	FIA - 1. El sistema será capaz de realizar cualquier proceso con una tasa de fallo del 0.05%.
Robustez	Prevención de caídas del sistema.	ROB -1. El sistema será capaz de hacer funcionar una copia del sistema automáticamente si el sistema deja de funcionar.
Rendimiento	Carga del sistema	REN -1. El sistema será capaz de gestionar 1000 robots Da Vinci.

		REN -2. El sistema será capaz de soportar 2000 usuarios simultáneos.
	Tiempo de proceso	REN -3. El sistema será capaz de dar una respuesta a cualquier petición del usuario en menos de 10 segundos.
	Tiempo de respuesta de BD	REN -3. El sistema resolverá cualquier consulta, actualización, inserción o eliminación en la base de datos en menos de 1 segundo.
	Optimización	REN -4. El sistema
Interoperabilidad	Comunicación con los robots	INT - 1. El sistema podrá acceder a cada robot sin incompatibilidades.
	Comunicación entre sistemas	INT - 2. El sistema podrá acceder a todos los sistemas necesarios sin incompatibilidades.
Safety	Safety	SAF - 1. El sistema garantiza la seguridad física del usuario.
Modificabilidad	Fácilmente modificable	MOD - 1. Los componentes serán lo más sencillos posible para una fácil modificación.
Usabilidad	Fácil de usar	USA - 1. El usuario podrá usar cualquier funcionalidad del sistema en menos de 5 clicks.
	Fácil de aprender	USA - 2. El usuario podrá aprender a usar el sistema con un sencillo tutorial de 1 hora.
Mantenibilidad	Fácil de mantener	MAN - 1. El sistema será capaz de hacer el mantenimiento en una hora.
Manejabilidad	Fácil de administrar	MANE - 1. El sistema será ajustable por el usuario con menos de 3 pantallas distintas.
Verificabilidad	Fácil de testear	VER-1. Un ingeniero destinado a realizar pruebas al sistema será capaz de configurar qué resultados de ejecución se registrarán durante dichos test.
	Fácil de verificar	VER-2. El sistema se podrá verificar satisfactoriamente en menos de 10 pruebas.
Reusabilidad	Reciclaje de componentes	REU-1. El 30% de la arquitectura

		será reutilizable para futuras aplicaciones del mismo ámbito.
Extensibilidad	Capacidad de expandir la funcionalidad del sistema.	EXT - 1. El sistema podrá ser aplicado en otras máquinas similares siguiendo unos pasos de configuración con duración inferior a dos horas.
Eficiencia	Optimización	EFI -1. El 35% de la capacidad del procesador y la memoria se mantendrán libres para los picos de procesamiento que puedan suceder.

4.- Vistas arquitectónicas

4.1.- Vista lógica

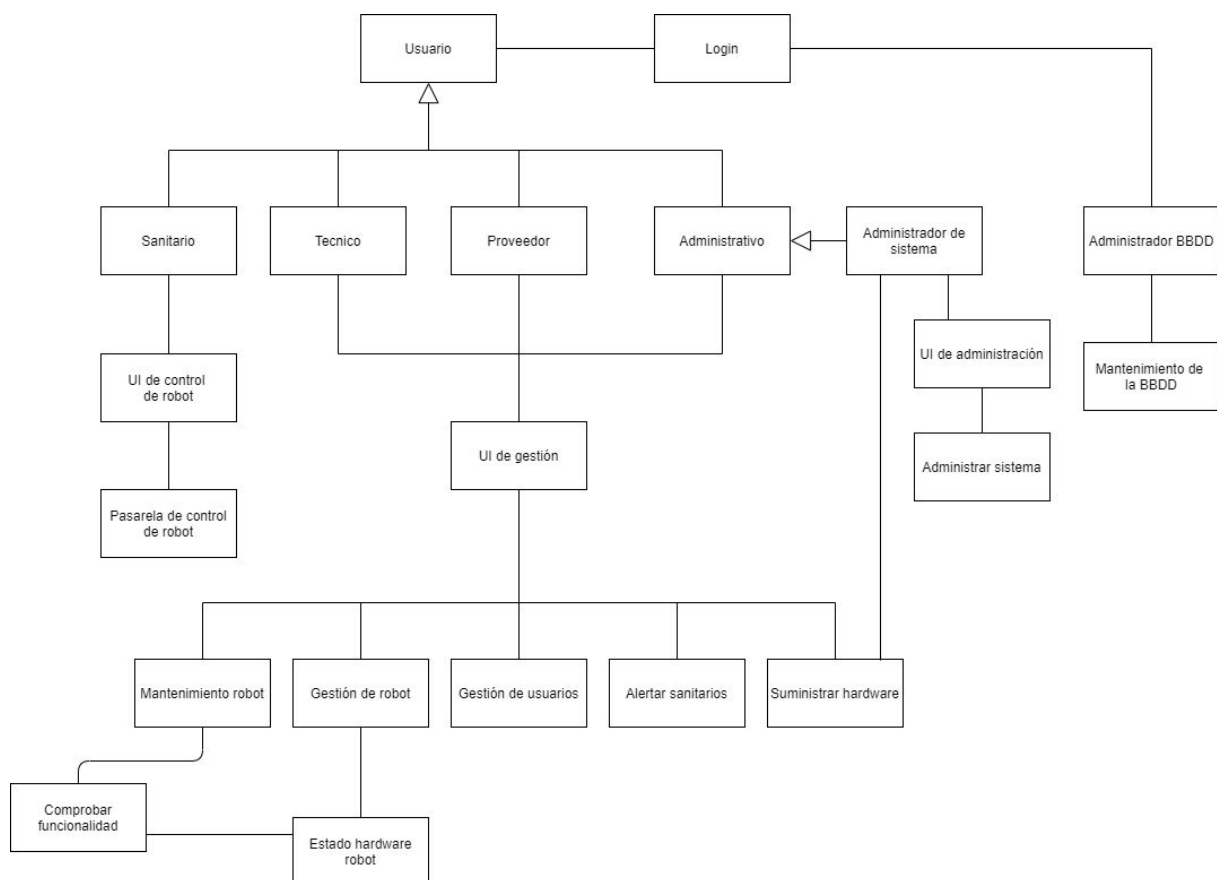
4.1.1.- Descripción

En esta vista vemos representados los requisitos funcionales como módulos, dichos módulos representan la estructura del diagrama de clases a implementar.

4.1.2.- Notación

Cada rectángulo representa una clase con su nombre, unidos por las relaciones entre ellas.

4.1.3.- Vista



4.1.4.- Catálogo

Para usar la aplicación debes tener una cuenta ("Login"), hay distintos tipos de usuarios que pueden tener acceso: "Sanitario" el médico que va a realizar la operación, a través de la interfaz "UI de control de robot" implementada por "Pasarela de control de robot" se podrá conectar y controlar a un robot determinado. "Técnico" encargado de comprobar el estado funcional de los robots. "Administrativo" es el que coordina las operaciones. Asigna y avisa a los "Sanitarios". Además, gestiona los "Sanitarios" y los robots registrados. Este módulo tiene una relación de herencia con "Administrativo del sistema", a través de la interfaz "UI de administración" implementada por "Administrar sistema" revisa y gestiona los

administrativos y los suministros de los proveedores. “Proveedor”, mediante la aplicación se lleva un registro de las piezas suministradas por los distintos proveedores. También se registran las peticiones y los envíos. Cada tipo de usuario implementa la interfaz “UI de gestión” con las funcionalidades que necesitan. Por último, “Administrador de BBDD” encargado del mantenimiento de la base de datos de la aplicación, no necesita tener una cuenta por eso no forma parte de la herencia de los usuarios.

4.1.5.- Justificación/Rationale (impacto de los QAs en la vista)

Los atributos de calidad que nos han llevado a tomar ciertas decisiones en el diseño del diagrama de clases son:

Control de estado del robot, es de las funcionalidades más importantes de nuestra plataforma tenemos en cuenta que solo las personas autorizadas pueden acceder a esta información.

Fácilmente modificable, fácil de administrar, fácil de testear, fácil de verificar, por ello nuestras clases cumplirán la ley de encapsulamiento.

4.2.- Vista de procesos

4.2.1.- Descripción

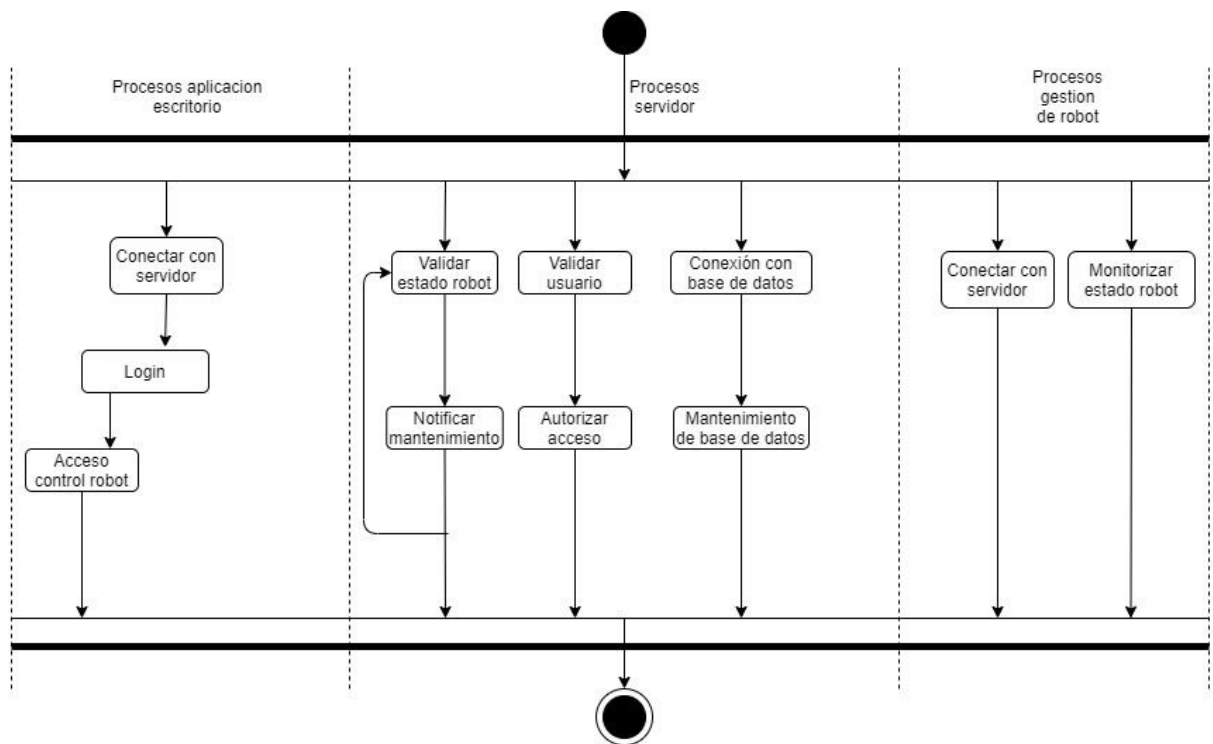
En esta vista se representan los distintos procesos que ocurren de forma constante durante la ejecución del sistema.

4.2.2.- Notación

Se representa mediante un diagrama de flujo. El punto negro superior indica el inicio del sistema, mientras que el inferior representa el final de la ejecución del sistema.

Las cajas representan las diversas acciones que se van realizando y las flechas el orden que llevan dichas acciones.

4.2.3.- Vista



4.2.4.- Catálogo

Los procesos de aplicación de escritorio son los que permiten a un usuario médico acceder al control de una de las máquinas desde cualquier lugar donde esté habilitada la aplicación, para lo cual primero se conectará con el servidor, luego el usuario introducirá sus credenciales de login para poder acceder y finalmente, tras la comprobaciones oportunas, se le dará acceso al control del robot.

Los procesos del servidor son los que están siendo realizados constantemente por el sistema para poder garantizar que todo esté disponible en todo momento. Por un lado, está constantemente validando el estado de los robots, para poder notificar la necesidad de mantenimiento si alguna parte fallase, para volver a la validación una vez ese mantenimiento se ha realizado. Por otro lado, cuando recibe un login, debe validar al usuario antes de autorizar el acceso. Finalmente, tiene que mantener una conexión constante con la base de datos para poder garantizar la mayor rapidez de actuación, y además poder saber rápidamente cuando algo falla y avisar a mantenimiento para que lo arregle.

Por último, los procesos de la gestión del robot, que se realizan cuando se accede al control de los robots. Deben mantener una constante conexión con el servidor para garantizar una conexión ininterrumpida entre el médico y el robot, que podría resultar fatal para un paciente en caso de perderse. También dispone de un seguimiento del estado de las diversas partes del robot, principalmente del material médico y de los diversos sensores utilizados para medir valores delicados.

4.2.5.- Justificación/Rationale (impacto de los QAs en la vista)

La relación entre los procesos y los atributos de calidad es para con esos atributos que hagan referencia a cosas que deben ocurrir de forma constante, manteniéndose un correcto funcionamiento, como temas de seguridad, de control del robot, de datos y de comunicación. Similarmente está relacionado con los business goals relacionados con los mencionados tópicos, principalmente los 1, 2, 3, 4 y 6.

4.3.- Vista de desarrollo

4.3.1.- Descripción

En esta vista se representa los diferentes componentes software que están agrupados en un mismo paquete de código ya sea por sus relaciones en cuanto a lo que se pretende que hagan o por que tengan dependencias con los mismos componentes.

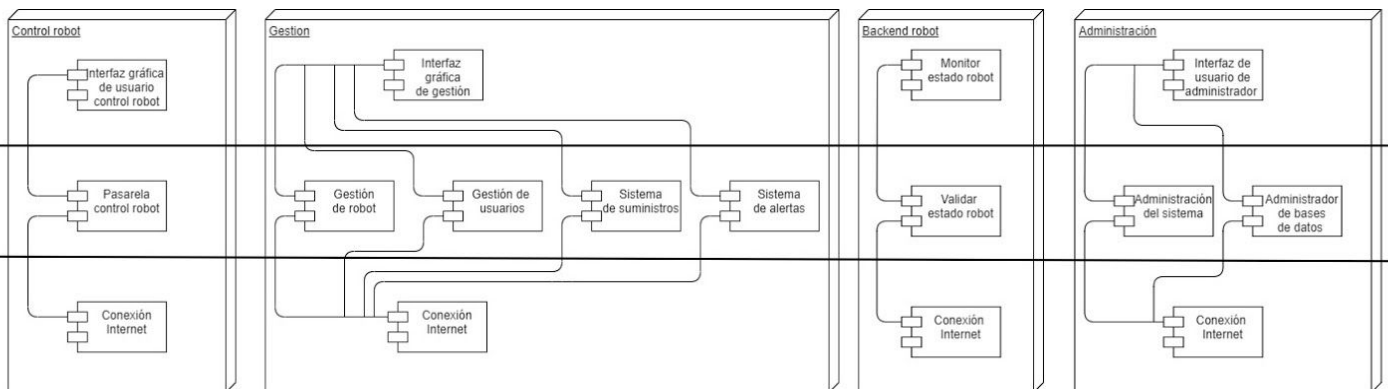
4.3.2.- Notación

Los rectángulos tridimensionales representan los paquetes de software.

Las cajas con conectores representan los componentes software.

Las líneas son las relaciones entre ellos.

4.3.3.- Vista



4.3.4.- Catálogo

Paquete control de robot: Contiene toda la funcionalidad destinada a acceder al control del robot.

Componente interfaz gráfica de usuario de control de robot: Es la interfaz gráfica través de la cual se accede al control de un robot.

Componente pasarela control robot: Es la parte del software que permite al personal sanitario entrar al control del robot autenticándole en el proceso, a través del componente conexión a internet.

Componente conexión a internet: Es el componente de software encargado de generar los paquetes de información que se enviarán a través de internet para el uso de la funcionalidad.

Paquete de gestión: Contiene toda la funcionalidad relacionada con la gestión que se hará de los robots, los usuarios del sistema, los suministros, y las alertas.

Componente interfaz gráfica usuario de gestión: Es la interfaz gráfica través de la cual se accede a los elementos de gestión de los robots, usuarios del sistema, las alarmas, y los suministros.

Componente gestión de robot: Contiene toda la funcionalidad de gestión de los robots registrados en el sistema como por ejemplo su disponibilidad de uso, la monitorización de su estado, programación de su mantenimiento, etc.

Componente de gestión de usuarios: Contiene toda la funcionalidad de gestión de usuarios del sistema como por ejemplo su registro, autenticación, eliminación, asignación de robots a sanitarios, etc.

Componente de sistema de suministros: Contiene toda la funcionalidad necesaria para que se puedan hacer peticiones a proveedores y estos las realicen.

Componente de sistema de alarmas: Contiene toda la funcionalidad necesaria para que se puedan enviar alarmas de distintos tipos como la de notificación a un sanitario para que use un robot.

Componente conexión a internet: Es el componente de software encargado de generar los paquetes de información que se enviarán a través de internet para el uso de la funcionalidad.

Paquete de backend robot: Contiene toda la funcionalidad relacionada con monitorizar y validar el estado del robot.

Componente monitor estado robot: Permite visualizar el estado del robot desde dentro del propio robot.

Componente validar robot: Contiene la funcionalidad para validar el estado físico del robot mediante pruebas rutinarias y enviar los resultados por internet.

Componente conexión a internet: Es el componente de software encargado de generar los paquetes de información que se enviarán a través de internet para el uso de la funcionalidad.

Paquete de administración: Contiene toda la funcionalidad relacionada con la administración del sistema.

Componente interfaz de usuario administrador: Es la interfaz gráfica a través de la cual se accede a la funcionalidad de administración del sistema y de la base de datos.

Componente de administración del sistema: Contiene la funcionalidad de administración del sistema como, reiniciar el sistema, cambiar valores de variables del sistema, registrar usuarios, robots, proveedores, gestionar caídas del sistema, etc.

Componente de administración de base de datos: Contiene toda la funcionalidad para monitorear la base de datos, realizar las labores de mantenimiento y todo lo necesario para su correcto funcionamiento.

Componente conexión a internet: Es el componente de software encargado de generar los paquetes de información que se enviarán a través de internet para el uso de la funcionalidad.

4.3.5.- Justificación

Con este diseño se espera conseguir los objetivos de mantener los datos confidenciales, íntegros y disponibles durante toda la vida útil del sistema. Además se espera que el rendimiento y eficacia del sistema sea óptimo y no ralentice o sobrecargue los servidores. Manteniendo el sistema en funcionamiento sin que falle

y recuperándose de los fallos en caso de que ocurran. Para terminar, permitir al sistema ser fácilmente portable a otros entornos, además de fácil de testear y verificar.

4.4.- Vista de despliegue

4.4.1.- Descripción

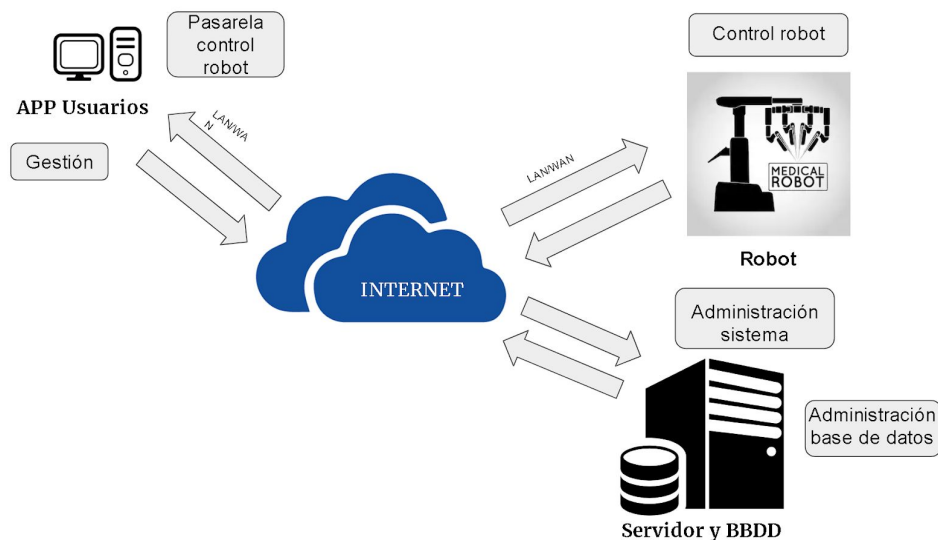
En esta vista se muestra el diagrama de despliegue de la aplicación. En él se muestran los elementos que forman parte del mismo y las conexiones entre ellos.

4.4.2.- Notación

Se representa cada elemento con una representación en modo esquemático. Cada flecha representa conexión entre los elementos y se especifica qué tipo de conexión.

4.4.3.- Vista

Diagrama de despliegue:



4.4.4.- Catálogo

El esquema muestra cuatro puntos importantes:

El equipo desde el que el usuario realiza las tareas de gestión y accede a las pasarelas para poder utilizar los robots.

Los robots que se pretenden controlar.

El servidor que se encarga de gestionar las peticiones de administración y la base de datos.

Todos ellos conectados a través de Internet.

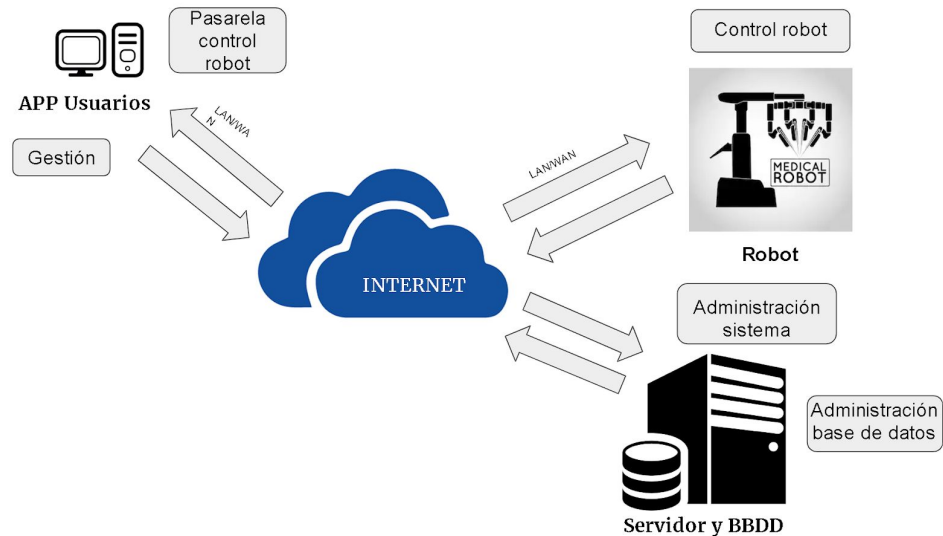
4.4.5.- Justificación/Rationale

Los atributos de calidad que han potenciado que el despliegue se realice como se ha indicado han sido:

Servicio continuo, segura, control del estado del robot, prevención de caídas, carga del sistema.

4.4.3.- Notación

Diagrama de despliegue:



4.4.4.- Uso

Se espera un rendimiento óptimo de la aplicación en cada equipo, debido a la simplicidad y eficiencia del código desplegado en los mismos.

Para la instalación del software, se entregará a los usuarios un ejecutable (previa comprobación de la veracidad de la información personal del usuario) que ejecutará en su equipo. Dicho ejecutable, accionará un wizard que guiará al usuario hasta completar la instalación.

El software tendrá una comprobación automática de versiones, por si se incluyen nuevas features, que pueda actualizarse para mantenerse lo más al día posible con los avances que vayan surgiendo en el mundo de la medicina.

4.5.- Escenarios

Esta vista consiste en un subconjunto de escenarios importante como son los casos de uso, con intención de mostrar cómo los elementos de las 4 vistas anteriores trabajan de manera conjunta.

4.5.1.- Descripción

Hemos considerado 6 actores en el escenario.

- En primer lugar, tenemos el administrador del sistema, el cual tiene acceso a la administración del propio sistema, y se encarga de la gestión de los usuarios del sistema. Por otra parte, también tiene acceso a las funciones de los administrativos del hospital, los cuales serán descritos más adelante.

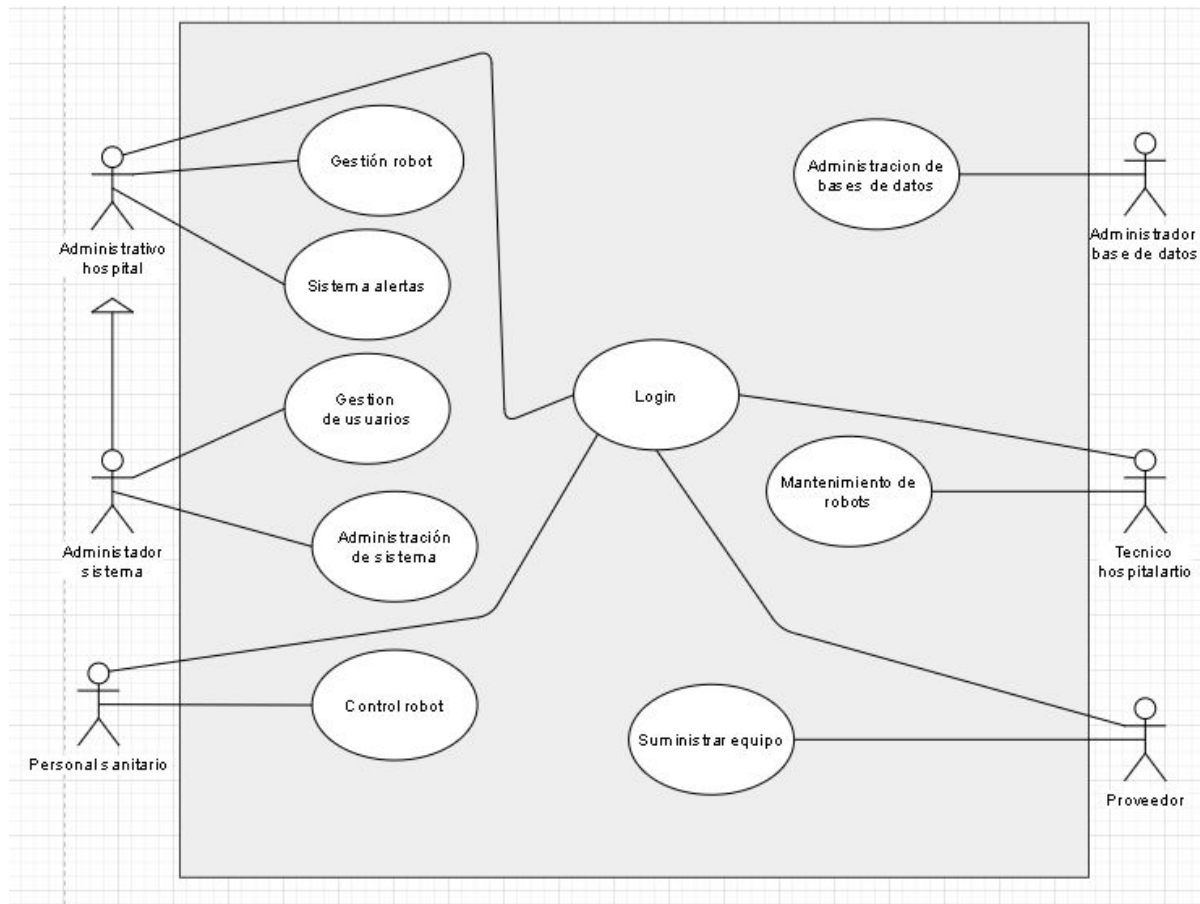
- El administrativo del hospital se encarga del sistema de alertas, el cual avisa al personal sanitario en caso de necesidad y de gestionar los robots, ya sea añadiendo en caso de recibir nuevo material, o eliminando alguno que se encuentre inactivo en caso de cualquier tipo de problema o necesidad de mantenimiento.
- El personal sanitario se encargará del control del robot exclusivamente, ya que es el encargado de utilizar el mismo para las intervenciones.
- El administrador de la base de datos tiene como función la dicha por su propio nombre, administrar y mantener las bases de datos referentes a nuestro sistema.
- El técnico hospitalario se encargará de mantener el estado de los robots, para garantizar la precisión de los mismos, evitando que surja cualquier problema a la hora de realizar intervenciones a distancia.
- Finalmente, tenemos el proveedor, el cual tiene como objetivo suministrar el equipo técnico y médico necesarios para que los robots de nuestro sistema sean completamente funcionales y en caso de ser requerido, de sustituir piezas en un mantenimiento.

Cabe destacar que todos los actores identificados en el sistema a excepción administrador de la base de datos deberán estar identificados mediante un login para poder acceder a las funcionalidades del sistema, siendo previamente dados de alta por el administrador del sistema.

4.5.2.- Notación

En cuanto a la notación de la vista, hemos optado por utilizar la misma de StarUML, pero siendo dicho diagrama representado en la herramienta draw.io, la cual tiene la misma notación y posee los mismos elementos que los diagramas UML.

4.5.3.- Vista



4.5.5.- Catálogo

Identificamos 3 grupos de Stakeholders en los casos de uso.

- En primer lugar tenemos a aquellos que trabajan en el hospital, encargados de gestionar a los trabajadores de la misma, es decir, los usuarios; de gestionar las herramientas de trabajo de los especialistas, es decir, los robots; tanto de los encargados de utilizar dichos dispositivos para realizar las intervenciones. Para ello, se utilizará un sistema de alertas, el cual tiene como objetivo llamar a un especialista para realizar una operación utilizando los robots Da-Vinci que tenemos en el sistema.
- Por otra parte, tenemos al personal capacitado para el mantenimiento y cuidado del sistema, donde distinguimos al encargado de las bases de datos de nuestro sistema, realizando los mantenimientos necesarios para su correcto funcionamiento; y el técnico que realizará el mantenimiento de los dispositivos Da-Vinci para que puedan seguir en funcionamiento en el hospital.
- Finalmente tendremos a los proveedores, encargados de suministrar el equipo, el cual será pedido mediante solicitudes del material necesario faltante en el hospital, dejando constancia en las existencias del hospital (en su base de datos).

4.5.6.- Justificación/Rationale (impacto de los QAs en la vista)

Los atributos de calidad que más impacto han tenido en el desarrollo de esta vista han sido:

- Privacidad, integridad y confidencialidad (Security): a nivel de que todos los usuarios deben hacer login antes de poder acceder a cualquier funcionalidad del sistema, exceptuando el administrador de bases de datos.
- Comunicación con los robots (Interoperabilidad): el personal hospitalario podrá conocer el estado del robot en cualquier momento. Al igual que el personal sanitario será el capacitado para el control de los mismos.
- Control del estado del robot (Precisión): al igual que el apartado anterior, será de vital importancia conocer el estado del dispositivo previamente a realizar una intervención.
- Safety: se garantizará la seguridad física de los usuarios mediante el control permanente del estados de los dispositivos Da Vinci, siendo estrictamente necesario su buen funcionamiento y estado para realizar una operación.

5.- Trazabilidad

5.1.- Entre vistas

Despliegue/ Escenarios

Despliegue/Escenarios	Gestión de robot	Login	Sistema de alertas	Gestión de usuarios	Admin. de sistema	Suministrar equipo	Admin. BBDD	Mantenimiento de robots	Control de robot
APP Usuario	X	X	X	X	X	X	X	X	X
Robot	X							X	X
Servidor y BBDD	X	X	X	X	X	X	X	X	X
Internet		X	X			X			X

Proceso/Escenarios

Proceso/Escenarios	Gestión de robot	Login	Sistema de alertas	Gestión de usuarios	Admin. de sistema	Suministrar equipo	Admin. BBDD	Mantenimiento de robots	Control de robot
--------------------	------------------	-------	--------------------	---------------------	-------------------	--------------------	-------------	-------------------------	------------------

Conectar con servidor	X	X	X	X	X	X	X		X
Login		X							
Acceso de control de robot									X
Validar estado robot	X							X	
Notificar mantenimiento	X		X					X	
Validar usuario				X					
Autorizar acceso		X		X					
Conexión con BBDD					X		X		
Mantenimiento de BBDD							X		
Monitorizar estado robot	X							X	

Implementación/Escenarios

Implementación /Escenarios	Gestión de robot	Login	Sistema de alertas	Gestión de usuarios	Admin. de sistema	Suministrar equipo	Admin. BBDD	Mantenimiento de robots	Control de robot
Interfaz gráfica usuario control de robot									X
Pasarela control de robot									X
Conexión de internet	X	X	X	X	X	X	X	X	X
Interfaz gráfica de gestión	X		X	X		X			
Gestión de robot	X								
Gestión de usuarios				X					
Sistema de						X			

suministros									
Sistema de alertas			X						
Monitor estado robot	X							X	
Validar estado robot	X							X	
Interfaz de usuario de administrador					X		X		
Administración del sistema					X				
Administrador de bases de datos							X		

Lógica/ Escenarios

Lógica/Escenarios	Gestión de robot	Login	Sistema de alertas	Gestión de usuarios	Admin. de sistema	Suministrar equipo	Admin. BBDD	Mantenimiento de robots	Control de robot
Login		X							
Usuario		X							
Sanitario		X							X
UI control robot									X
Pasarela de control de robot									X
Técnico		X						X	
Mantenimiento robot								X	
Comprobar funcionalidad								X	
Proveedor		X				X			

Suministrar hw						X			
Administrador BBDD							X		
Mantenimiento de la base de datos							X		
Administrativo	X	X	X	X		X			
UI administración					X				
UI gestión	X			X					
Administrador de sistema		X			X				
Administrar sistema	X		X	X	X				
Gestión de robot	X								
Estado hw robot			X					X	
Alertar sanitarios			X						
Gestión de usuarios				X					

Proceso/Despliegue

Proceso/Despliegue	APP Usuario	Robot	Servidor y BBDD	Internet
Conectar con servidor	X	X	X	X
Login	X			X
Acceso de control de robot		X		
Validar estado robot		X		
Notificar mantenimiento	X			X
Validar usuario	X			X
Autorizar acceso	X		X	X
Conexión con BBDD			X	
Mantenimiento de BBDD			X	
Monitorizar estado robot		X		

Implementación/Despliegue

Implementación /Despliegue	APP Usuario	Robot	Servidor y BBDD	Internet
Interfaz gráfica usuario control de robot	X	X		
Pasarela control de robot		X	X	X
Conexión de internet			X	X
Interfaz gráfica de gestión	X			
Gestión de robot	X	X	X	X
Gestión de usuarios	X		X	X
Sistema de suministros	X		X	X
Sistema de alertas	X			X
Monitor estado robot		X		
Validar estado robot		X	X	
Interfaz de usuario de administrador	X			
Administración del sistema	X	X	X	X
Administrador de bases de datos			X	X

Lógico/Despliegue

Lógico/Despliegue	APP Usuario	Robot	Servidor y BBDD	Internet
Login	X		X	X
Usuario	X			
Sanitario	X	X	X	X
UI control robot	X	X		
Pasarela de control de robot		X	X	X
Técnico		X		
Mantenimiento robot		X		
Comprobar funcionalidad		X		
Proveedor	X			X
Suministrar hw	X		X	X
Administrador BBDD			X	X
Mantenimiento de la base de datos			X	X

Administrativo	X			X
UI administración	X			
UI gestión	X			
Administrador de sistema	X		X	X
Administrar sistema	X		X	X
Gestión de robot	X	X	X	X
Estado hw robot		X	X	X
Alertar sanitarios	X		X	X
Gestión de usuarios	X		X	X

Implementación/Proceso

Implementación /Proceso	Conectar con servidor	Login	Acceso control robot	Validar estado robot	Notificar mantenimiento	Validar usuario	Autorizar acceso	Conexión BBDD	Mantenimiento de BBDD	Monitorizar estado robot
Interfaz gráfica usuario control de robot		X	X							
Pasarela control de robot	X		X							
Conexión de internet	X	X	X		X		X	X	X	
Interfaz gráfica de gestión		X								
Gestión de robot	X			X	X					X
Gestión de usuarios	X									
Sistema de suministros	X									
Sistema de alertas	X									
Monitor estado robot										X
Validar estado robot				X	X					X

Interfaz de usuario de administrador		X								
Administración del sistema	X					X	X			
Administrador de bases de datos	X							X	X	

Lógico/Proceso

Lógico/ Proceso	Conectar servidor	Login	Acceso control robot	Validar estado robot	Notificar mantenimiento	Validar usuario	Autorizar acceso	Conexión BBDD	Mantenimiento de BBDD	Monitorizar estado robot
Login		X								
Usuario		X								
Sanitario		X	X			X	X			
UI control robot			X							
Pasarela de control de robot			X							
Técnico		X		X	X	X	X			X
Mantenimiento robot				X	X					X
Comprobar funcionalidad				X						X
Proveedor	X	X				X	X			
Suministrar hw	X									
Administrador BBDD								X	X	
Mantenimiento de la base de datos								X	X	
Administrativo		X			X	X	X			
UI administración						X	X			
UI gestión					X					X
Administrador		X				X	X			

de sistema										
Administrar sistema	X									
Gestión de robot	X			X						X
Estado hw robot	X			X						X
Alertar sanitarios	X				X					
Gestión de usuarios	X					X	X			

Implementación/Lógico

Anexo 1.

5.2.- Entre Business Goals y vistas

Business Goals/Vistas	Escenarios	Despliegue	Procesos	Implementación	Lógica
BG-0	X				
BG-1			X	X	X
BG-2		X	X		
BG-3		X	X	X	
BG-4		X	X	X	
BG-5	X				
BG-6	X	X	X	X	X

- BG-0: El sistema debe subsistir durante al menos 10 años sin necesidad de cambiar nada a nivel de la arquitectura, por lo tanto esto será representado por la vista de escenarios, por lo que en dicha vista, veremos el funcionamiento del sistema durante dicho periodo de tiempo.
- BG-1: Para cumplir este objetivo de negocio, los procesos que se realizan deben garantizar la integridad y seguridad de los datos que se manejan. Lo que afecta también a su estructura lógica y cómo se implementará.
- BG-2: Para garantizar el funcionamiento ininterrumpido del sistema (exceptuando en mantenimientos, que deberán ser siempre a unas horas tabuladas), la vista de procesos está identificada principalmente, debido a que está plenamente relacionada con el runtime, y el funcionamiento de los threads. Por otra parte a nivel

de despliegue, tenemos que tener en cuenta la Performance del sistema, la cual se encuentra identificada en el despliegue del mismo.

- BG-3: Para conseguir unos datos precisos, en la implementación del sistema deberá perseguir unos estándares de precisión. A nivel de procesos, estos deberán comunicar al sistema los valores necesarios correctamente. Finalmente, en cuanto al despliegue, todo deberá estar conectado debidamente para evitar errores de cálculo.
- BG-4: La rápida respuesta del sistema está directamente relacionada con los procesos, donde mediante la implementación del sistema, se busca tener la velocidad de respuesta más rápida, independientemente del número de usuarios que haya conectados simultáneamente. Por lo tanto, a la hora del despliegue será importante conseguir que el sistema no se caiga debido a un alto nivel de carga.
- BG-5: Los escenarios tratan de ilustrar la arquitectura, por lo que es de vital importancia que los usuarios de nuestro sistema entiendan el funcionamiento del mismo para poder utilizarlo lo más pronto posible.
- BG-6: La ampliación del sistema está relacionado principalmente con la implementación y lógica del sistema, donde el mismo debe ser pensado y preparado para una futura expansión. Por otra parte, forma parte del despliegue y de los procesos, debido a que son las partes de la arquitectura más afectadas por los cambios realizados a la hora de realizar un aumento de la capacidad del sistema. Finalmente, ya que nos encontramos bajo el sistema de vistas 4+1, al formar parte de todas las anteriores, también aparecerá en los escenarios, donde esta vista demuestra como las 4 anteriores funcionan de manera conjunta.

5.3.- Entre atributos de calidad y vistas

Atributos de calidad/ Vistas	Escenarios	Despliegue	Procesos	Implementación	Lógica
Servicio continuo	X	X	X		
Privacidad	X		X	X	X
Integridad	X		X	X	X
Confidencialidad	X		X	X	X
Control de estado del robot	X		X		X
Prevención de fallos	X			X	
Prevención de caídas de sistema	X	X			
Carga del sistema	X		X	X	X
Tiempo de proceso.	X		X	X	X
Interfaces externa	X			X	X

Interoperabilidad	X			X	X
Safety	X	X			
Datos correctos	X		X	X	
Tiempo de respuesta de BD.	X		X		
Comunicación entre sistemas	X	X	X		
Fácilmente modificable	X			X	X
Optimizado	X	X		X	X
Fácil de usar	X				
Fácil de aprender	X				
Fácil de mantener	X	X		X	X
Fácil de administrar	X				
Fácil de testear	X	X			X
Fácil de portar	X			X	X
Fácil de verificar	X	X			X
Reciclaje de componentes	X			X	X
Capacidad de expandir la funcionalidad del sistema	X	X		X	X

La vista de escenarios tiene todos los atributos de calidad ya que representa el sistema al completo y tiene que verificar todos los atributos de calidad.

La vista de despliegue es el diseño de cómo se va a desplegar el sistema a nivel físico y tiene que permitir un servicio continuo sin interrupciones, ser capaz de prevenir caídas de sistema, asegurar la comunicación entre los distintos sistemas, que sea fácil de testear, verificar, mantener y expandirse , además que este optimizado para asegurar su velocidad y la seguridad de los componentes hardware.

La vista de proceso es el diseño de cómo la información va a fluir el sistema por lo que tiene que asegurar su funcionamiento continuo, la seguridad de los datos y que estos sean fiables. Además tiene que asegurar el buen rendimiento del sistema y que la comunicación con otros sistema se produzca sin problemas.

La vista de implementación es el diseño de cómo los distintos paquetes software se comunican entre ellos por lo que tiene que asegurar la seguridad de los datos, el

rendimiento óptimo de todo el sistema, la correcta comunicación entre las interfaces, y ser fácil de testear, verificar, mantener, expandir y cambiar cuando sea necesario.

La vista lógica es el diseño de cómo cada elemento de los paquetes de software se conecta con el resto del sistema y una descripción de su función por lo que tiene que asegurar la seguridad de los datos y su fiabilidad, además tiene que estar bien construido para asegurar el buen rendimiento de todo el sistema. Esta parte también interviene en la comunicación con otros sistemas e interfaces, por lo que necesita asegurar su correcto funcionamiento. También debe ser fácil de testear, verificar, mantener, expandir y cambiar cuando sea necesario.

6.- Conclusiones

6.1.- Relativas a la arquitectura

Tras el estudio de los objetivos de negocio y los atributos de calidad, el equipo decidió que una posible solución sería utilizar una arquitectura por capas. Con el patrón por capas, se propone modularizar todo por componentes de manera que cada utilidad necesaria de la aplicación esté separada del resto.

De los objetivos de negocio se puede entender que el software debe ser usable, seguro y flexible.

Con un patrón por capas, podemos separar las interfaces de la lógica, lo que favorece la simplicidad a la hora de aprender a utilizar la aplicación. Cada usuario aprende sólo el componente que necesita para desempeñar su función en el entorno laboral.

La separación por capas, permite también una mejor protección de los datos, ya que al estar separada la lógica de acceso a ellos, se puede prevenir inyección de datos de manera ajena a la aplicación de manera más óptima y fiable.

Al estar los componentes separados, es mucho más sencillo la búsqueda y solución de errores, lo que ayuda a que el sistema se mantenga siempre activo. Al detectarse un error, los desarrolladores pueden filtrar por componentes dónde se encuentra y aislar dicho componente para solucionarlo, sin necesidad de que el resto del sistema se vea afectado.

A la hora del despliegue, se tendrá una primera instalación con una carga computacional elevada, debido a la cantidad de componentes separados de los que dispone el sistema. Una vez instalado, las actualizaciones serán mucho más ligeras y sin apenas carga, ya que toda incorporación que se haga se añadirá a partir de un componente nuevo, que no afecta a los ya existentes ni al uso normal de la herramienta.

Por el mismo motivo, los desarrolladores tendrán mucha facilidad para la incorporación de nuevas features en el futuro.

Aparte de todos los beneficios de cara a cumplir con los objetivos de negocio marcados, éste tipo de patrón favorece la arquitectura cliente-servidor, también necesaria para un mejor desempeño de las funciones ofrecidas por la herramienta para la monitorización de los robots incluidos en el sistema.

6.2.- Personales

En esta práctica hemos mejorado nuestras capacidades de documentación de una arquitectura y de patrones de diseño. Hemos adquirido los conocimientos de las distintas vistas arquitectónicas.

Por otra parte, nos hemos encontrado con una falta de documentación teórica para afrontar las distintas secciones de la práctica. Por ejemplo, en la sección de las vistas, en algunos diagramas hemos tenido serias dificultades para sacarlos adelante, debido a que en las transparencias de la asignatura solo había un ejemplo sin explicaciones adicionales de cómo se realiza dicho diagrama.

7.- Anexo 1

La tabla Lógica/Implementación está incluida en un archivo excel dentro del fichero .zip.

8.- Anexo 2

Se incluye en la carpeta .zip de la entrega todos los archivos en formato imagen de los diagramas realizados en el apartado de las vistas arquitectónicas por si surgiera un problema de visualización de los mismos en su versión .png (encontrados en la subcarpeta Vistas).