

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL



Facultad de Ingeniería en Electricidad y Computación

**“IMPLEMENTACIÓN DE UN SISTEMA DE SOPORTE MÉDICO PARA LA
SOCIEDAD DE TRAUMA DEL ECUADOR, CAPÍTULO
CHIMBORAZO”**

TRABAJO DE TITULACIÓN

Previa a la obtención del Título de:

MAGISTER EN SISTEMAS DE INFORMACIÓN GERENCIAL

Autor:

ING. LUIS EDUARDO CABEZAS PARRA

Guayaquil – Ecuador

Año: 2018

AGRADECIMIENTO

Doy gracias en primer lugar a Dios, por las bendiciones recibidas y por permitirme llegar a esta meta profesional.

Agradezco a mis padres, por darme la oportunidad de tener este crecimiento en mi vida profesional y por enseñarme que con esfuerzo y dedicación uno puede alcanzar sus metas.

A mi esposa, por el incondicional apoyo, amor y paciencia que me brindó todo este tiempo.

A mi hermana, por todos sus ánimos.

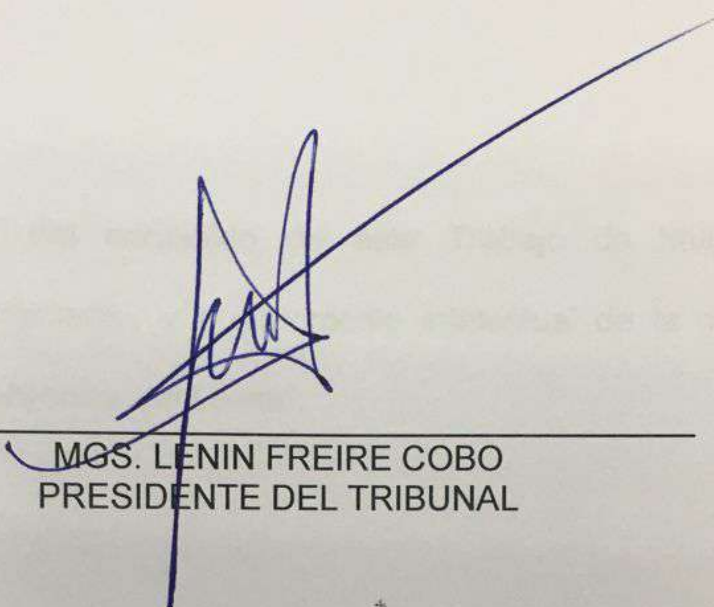
Por último, a la Sociedad de Trauma del Ecuador y sus integrantes, cuya colaboración fue esencial para el desarrollo de esta tesis.

DEDICATORIA

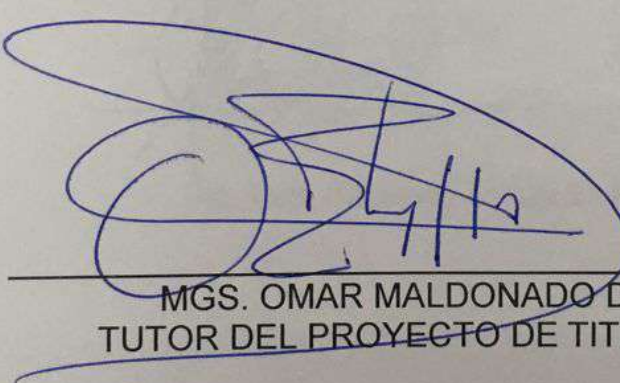
Dedico esta tesis a mis padres, a mi hermana y a mi esposa. En especial a mis padres, Eduardo Cabezas y Rocío Parra, que desde el inicio de este nuevo paso me han brindado su infinito apoyo, preocupación y motivación; por su amor y consejos, que siempre han sido la guía para mi vida.

A mi esposa Evelyn Delgado, mi fiel compañera de vida, por el tiempo y amor que ha invertido en mí para incentivar me en seguir luchando hasta llegar a este nuevo logro.

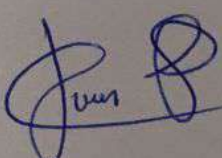
TRIBUNAL DE SUSTENTACIÓN



MGS. LENIN FREIRE COBO
PRESIDENTE DEL TRIBUNAL



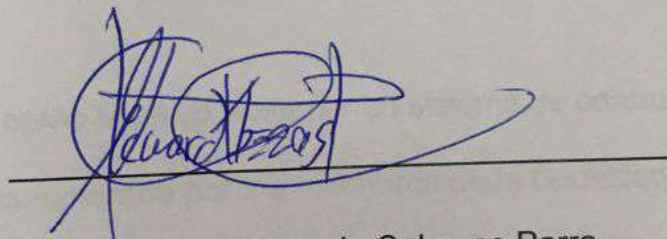
MGS. OMAR MALDONADO DAÑIN
TUTOR DEL PROYECTO DE TITULACIÓN



MGS. JUAN CARLOS GARCIA
MIEMBRO DEL TRIBUNAL

DECLARACIÓN EXPRESA

“La responsabilidad del contenido de este Trabajo de titulación, me corresponde exclusivamente; y el patrimonio intelectual de la misma a la Escuela Superior Politécnica del Litoral”.



Ing. Luis Eduardo Cabezas Parra

RESUMEN

El presente proyecto de titulación pretende otorgar una adecuada solución tecnológica para la Sociedad de Trauma del Ecuador – Capítulo Chimborazo, para la futura implementación de arquitecturas informáticas que brinden seguridad a la información.

Es trabajo tiene como objetivo proponer un sistema de control y administración de la información generada por los miembros de la Sociedad de Trauma, para el mejoramiento de los procesos existentes y la satisfacción de los pacientes. La propuesta nace a partir de la falta de herramientas informáticas comunes entre los miembros que permitan el control de su información en conjunto, con ese enfoque se procede a realizar el siguiente documento bajo la siguiente estructura:

Capítulo 1, se analizaron las generalidades, antecedentes, descripción del problema y análisis de las causas por las cuales se propone la implementación

de esta solución tecnológica. Se revisarán los objetivos específicos y la metodología a utilizar para el desarrollo de la propuesta.

Capítulo 2, está compuesto por el marco teórico en el cual se detallan los fundamentos de las arquitecturas tecnológicas, definiciones, términos y elementos en las que se apoyará el desarrollo de este trabajo.

Capítulo 3, detalla el estado actual de la empresa, en donde se realiza el respectivo levantamiento de la información con la finalidad de obtener los antecedentes, los procesos a optimizar y los requisitos necesarios para realizar las tareas diarias del personal.

Capítulo 4, profundiza en las estructuras realizadas de las arquitecturas que se utilizarán para el desarrollo de la solución propuesta, junto con el diseño de

los componentes que formarán parte de las mismas y los diferentes métodos que estarán disponibles para su acceso y ejecución.

Capítulo 5, explica las herramientas que serán necesarias para el desarrollo de los sistemas que conformarán la solución tecnológica y las técnicas de seguridad que se aplicarán a la misma, junto con las diferentes pruebas que se realizarán para comprobar la ejecución correcta de las funcionalidades ofrecidas.

Capítulo 6, analiza los resultados obtenidos en los puntos de mayor interés para la Sociedad de Trauma, siendo estos la mejora en tiempos de atención, seguridad de la información en resultados de exámenes de laboratorio y beneficios adicionales para los pacientes.

ÍNDICE GENERAL

AGRADECIMIENTO	ii
DEDICATORIA	iii
TRIBUNAL DE SUSTENTACIÓN	iv
DECLARACIÓN EXPRESA	v
RESUMEN.....	vi
ÍNDICE GENERAL	ix
ABREVIATURAS	xii
ÍNDICE DE FIGURAS.....	xiv
ÍNDICE DE TABLAS.....	xvii
INTRODUCCIÓN.....	xviii
CAPÍTULO 1.....	1
GENERALIDADES	1
1.1 Antecedentes.....	1
1.2 Descripción del problema	2
1.3 Solución propuesta	4
1.4 Objetivo General.....	7
1.5 Objetivos Específicos	7
1.6 Metodología.....	8
CAPÍTULO 2.....	9

MARCO TEÓRICO	9
2.1 Arquitectura Web Modelo-Vista-Template (MVT)	9
2.2 Arquitectura de acceso de datos REST API	14
2.3 Seguridad de la información mediante REST API y web tokens.....	24
2.4 Normativa de calidad ISO 13485.....	28
CAPÍTULO 3.....	37
SITUACIÓN ACTUAL Y REQUERIMIENTOS.....	37
3.1 Situación actual.....	37
3.2 Requerimientos funcionales	39
3.3 Procesos a optimizar	42
3.4 Actores y casos de flujo.....	45
CAPÍTULO 4.....	49
ANÁLISIS Y DISEÑO	49
4.1 Análisis de Web Services a implementar.....	49
4.2 Análisis de reportes necesarios.....	56
4.3 Diseño de interfaz gráfica.....	57
4.4 Diseño de base de datos.....	66
4.5 Diseño de reportes a implementar.....	66
CAPÍTULO 5.....	70
DESARROLLO Y PRUEBAS.....	70

5.1 Desarrollo de la aplicación REST	70
5.2 Desarrollo de la aplicación web	72
5.3 Pruebas internas	73
5.4 Pruebas de stress	80
5.5 Pruebas con usuarios.....	84
5.6 Implementación del sistema	91
CAPÍTULO 6.....	93
ANÁLISIS DE RESULTADOS.....	93
6.1 Análisis de tiempos utilizados en citas médicas antes y después de usar el sistema	93
6.2 Mejora en desempeño por parte de médicos.....	95
6.3 Mejora en satisfacción del servicio brindado a los pacientes	95
6.4 Análisis de cantidad de exámenes médicos de laboratorio alterados	104
CONCLUSIONES Y RECOMENDACIONES	106
BIBLIOGRAFÍA.....	109

ABREVIATURAS

API	:	Application Programming Interface
AS	:	Authorization Server
BMI	:	Body Mass Index
CAPA	:	Corrección Acción Prevención Acción
DNS	:	Domain Name Servers
HTTP	:	Hypertext Transfer Protocol
IP	:	Internet Protocol address
ISO	:	International Organization for Standarization
IT	:	Tecnologías de Información
JOSE	:	JSON Object Signing and Encryption
JSON	:	JavaScript Object Notation
JWE	:	JSON Web Encryption
JWS	:	JSON Web Signature
JWT	:	JSON Web Token

MVC	:	Modelo-Vista-Controlador
MVT	:	Modelo-Vista-Template
ORM	:	Object-Relational Mapper
REST	:	Representational State Transfer
SQL	:	Structured Query Language
URI	:	Uniform Resource Identifier
URL	:	Uniform Resource Locator
XML	:	eXtensible Markup Language

ÍNDICE DE FIGURAS

Figura 2.1: Funcionamiento de Infraestructura MVC	10
Figura 2.2: Funcionamiento de Infraestructura MVT	11
Figura 2.3: Estructura de JWT Encriptado	28
Figura 2.4: Procesos de Norma ISO 13485	29
Figura 3.1: Caso de Uso - Módulo Paciente	46
Figura 3.2: Caso de uso - Módulo Médico	47
Figura 3.3: Caso de Uso - Módulo Laboratorio	48
Figura 3.4: Caso de Uso - Módulo Sistemas	48
Figura 4.1: Estructura de interfaz gráfica	58
Figura 4.2: Vista de tipo de contenido – Texto.....	60
Figura 4.3: Vista de tipo de contenido – Información	61
Figura 4.4: Vista de contenido – Formulario	61
Figura 4.5: Vista de contenido – Gráficos	62
Figura 4.6: Vista de contenido – Tablas.....	63
Figura 4.7: Vista de contenido – Reporte.....	64

Figura 4.8: Vista de contenido – Agenda	64
Figura 4.9: Pantalla de ingreso al sistema	65
Figura 4.10: Esquema de Base de Datos	66
Figura 4.11: Creación de un reporte en el sistema	67
Figura 4.12: Estructura de ejecución para generar un reporte.....	68
Figura 5.1: Código de ejemplo de LocustIO.....	81
Figura 5.2: Pantalla de la aplicación LocustIO	82
Figura 5.3: Pantalla de resultados de LocustIO	83
Figura 6.1: Tiempos de atención (minutos) antes y después de aplicar el sistema	94
Figura 6.2: Encuesta realizada a los pacientes para medir nivel de satisfacción	96
Figura 6.3: Resultado de encuesta antes del sistema - Pregunta 1	97
Figura 6.4: Resultado de encuesta antes del sistema - Pregunta 2.....	97
Figura 6.5: Resultado de encuesta antes del sistema - Pregunta 3.....	98
Figura 6.6: Resultado de encuesta antes del sistema - Pregunta 4.....	98

Figura 6.7: Resultado de encuesta antes del sistema - Pregunta 5.....	99
Figura 6.8: Resultado de encuesta antes del sistema - Pregunta 6.....	99
Figura 6.9: Resultado de encuesta antes del sistema - Pregunta 7.....	100
Figura 6.10: Resultado de encuesta después del sistema–Pregunta 1	101
Figura 6.11: Resultado de encuesta después del sistema–Pregunta 2	101
Figura 6.12: Resultado de encuesta después del sistema–Pregunta 3	102
Figura 6.13: Resultado de encuesta después del sistema Pregunta 4	102
Figura 6.14: Resultado de encuesta después del sistema–Pregunta 5	103
Figura 6.15: Resultado de encuesta después del sistema–Pregunta 6	103
Figura 6.16: Resultado de encuesta después del sistema–Pregunta 7	104

ÍNDICE DE TABLAS

Tabla 1: Tipos de Conectores en Estructura de REST	15
Tabla 2: Roles de Componentes de Estructura REST	17
Tabla 3: Elementos de Datos de Estructura REST	18
Tabla 4: Requerimientos de Normativa ISO 13485.....	32
Tabla 5: Términos reservados para Módulos.....	51
Tabla 6: Términos reservados de tipos de acción.....	52
Tabla 7: Términos reservados para objetos.....	54
Tabla 8: Términos reservados de acciones	55
Tabla 9: Pruebas internas	74
Tabla 10: Pruebas con usuarios	85

INTRODUCCIÓN

La información de los pacientes es un elemento esencial y decisivo para los diagnósticos y la toma de decisiones de acciones a realizar para los médicos. Por lo que mantenerla segura y accesible es de gran importancia para poder realizar futuros análisis de la misma y el desarrollo de nuevas investigaciones y estudios para nuevos avances en la rama.

En el mundo actual de la medicina, el desarrollo de la tecnología y su continuo avance, ha afectado de forma significativa la manera que se desarrollan sus actividades diarias; debido a que el acceso inmediato e ininterrumpido que brinda la tecnología a la información, ha ayudado a que se acelere la velocidad para obtener nuevos descubrimientos y desarrollos en el ámbito de la medicina, y también en los procedimientos para de manera más rápida y efectiva poder brindar ayuda y soluciones a los pacientes.

A pesar de que actualmente existen soluciones tecnológicas que brindan beneficios y funcionalidades de ayuda para médicos, generalmente, ninguna cumple con todas las expectativas o requisitos que busca un médico para poder mejorar su desempeño laboral y el nivel de atención al paciente.

El brindar la información médica personal al paciente es un punto clave para cualquier médico, ya que esto brinda mayor seguridad y confianza al paciente y permite ofrecer una mejor satisfacción e interacción con el mismo.

La Sociedad de Trauma del Ecuador considera que es tiempo de avanzar junto con la tecnología para poder generar de una manera segura y atractiva, información que puede brindar beneficios tanto a los médicos miembros, como a los pacientes que son atendidos por los mismos.

CAPÍTULO 1

GENERALIDADES

1.1 Antecedentes

Dentro del proceso de análisis efectuado en las actividades diarias realizadas por los médicos integrantes de la Sociedad de Trauma del Ecuador - Capítulo Chimborazo, se pudo detectar que no existe ningún método común para poder administrar y compartir la información médica generada entre los integrantes de esta sociedad, ya que cada integrante utilizaba un sistema diferente para administrar su información; encontrando también que casi todos los distintos sistemas utilizados

carecían de políticas de seguridad para la información, actualizaciones y soporte técnico.

En algunos casos, estos sistemas no cuentan con un método de respaldo de la información, por lo que, en casos de daños en el equipo local, existía una pérdida irremediable de la información.

Actualmente existen una gran cantidad de sistemas médicos web, sin embargo, estos no generan la confianza necesaria en los integrantes de dicha sociedad, ya sea por no tener la autoría del sistema, no cumplir con algunos requisitos necesarios para el desarrollo de algunas tareas o falta de dichas tareas, o por miedo de perder la información ingresada en caso de problemas de pago de licencias.

También cabe recalcar que la entidad no cuenta con un Departamento de IT que ayude con las solicitudes, ni con el soporte que necesitan los integrantes de esta sociedad.

1.2 Descripción del problema

Actualmente, los diferentes médicos y entes médicos vinculados a la Sociedad de Trauma del Ecuador – Capítulo Chimborazo utilizan diferentes aplicativos y métodos para poder administrar su información interna, ninguno de estos teniendo algún vínculo entre sí, en algunos casos con defectos funcionales y sin contar con soporte técnico que garantice

correcciones a los mismos; o en otros casos utilizando aplicaciones de escritorio básicas y antiguas, sin tener la posibilidad de mejorar la aplicación o brindar nuevas funcionalidades.

Existe una molestia de ocurre de manera recurrente, la cual es la solicitud de información médica por parte de los pacientes de los médicos integrantes de esta sociedad; esto ocurre debido a que la solicitud hay que realizarla de manera personal y en muchas ocasiones los médicos no se encuentran presentes o disponibles, por lo que no hay una manera rápida de obtener esa información.

Actualmente, como no existe un vínculo entre la información generada por los médicos integrantes de la Sociedad de Trauma, entonces existen varios historiales médicos de los pacientes, cada uno diferente por médico, por lo que para poder obtener un historial médico completo y detallado, existe la necesidad de realizar múltiples solicitudes a los diferentes médicos que han atendido al paciente, volviendo a producirse el problema mencionado en el párrafo anterior.

En la actualidad, no existe ninguna forma de poder obtener retroalimentación por parte de los pacientes acerca del servicio brindado.

Por último, también han existido casos de resultados de laboratorio alterados y que han sido entregados a los médicos en las consultas por

parte de los pacientes, por lo que no existe seguridad de la información que es solicitada por parte del médico.

1.3 Solución propuesta

Los integrantes de la Sociedad de Trauma decidieron solicitar una solución para poder administrar su información interna y al mismo tiempo, poder generar un vínculo entre cada uno de ellos y los laboratorios médicos vinculados con dicha sociedad.

Solicitaron un sistema seguro en el que se pueda realizar el intercambio de información de pacientes relacionados, una mejor interactividad para solicitudes que realizan a los entes médicos para las cirugías a realizarse en dichos centros.

Se propone realizar un sistema de soporte médico que constará de cuatro módulos interrelacionados:

- Médico.
- Paciente.
- Laboratorio.
- Sistemas.

El sistema consistirá en una aplicación web, la cual tendrá una arquitectura responsive, que permitirá al usuario poder accederla desde diversos dispositivos móviles con comodidad visual.

Funcionalidades

Módulo Médico

Este módulo permitirá el ingreso y administración de la información médica de los pacientes. Esto se lo podrá realizar mediante la información generada mediante citas médicas, en las cuales se ingresarán los diferentes síntomas, diagnósticos, variación de datos clínicos (peso, altura, nivel de glucosa, entre otros); esto con la finalidad de poder generar un completo historial médico del paciente y que sea más fácil de analizar, mediante ayudas gráficas o reportes para poder determinar de mejor forma futuros diagnósticos. Con esta información también se ayudará al paciente mediante la generación automática de notificaciones o recordatorios de futuras citas, medicamentos a consumir, exámenes a realizarse, entre otros.

Módulo Paciente

Este módulo permitirá la actualización de los datos personales del paciente. También el paciente podrá consultar de manera inmediata su historial médico detallado, exámenes a realizarse (junto con el estado del

mismo en el laboratorio), medicamentos a consumir, futuras citas médicas y progreso en sus datos clínicos (peso, altura, nivel de glucosa, entre otros). Mediante el análisis de sus datos clínicos el sistema también mostrará recomendaciones para mejorar su salud basados en el análisis de dichos datos.

Por último, el paciente también podrá ingresar reclamos y recomendaciones como retroalimentación del servicio recibido.

Módulo Laboratorio

Las funcionalidades que brinda este módulo son:

- Consulta de solicitudes de exámenes solicitados por médicos.
- Actualización de estado de examen de laboratorio.
- Envío de resultados detallados de exámenes de laboratorio, pudiendo adjuntar archivos digitales de los exámenes de laboratorio.
- Consulta de historial de resultados de exámenes realizados.

Módulo de Sistemas

Las funcionalidades que brinda este módulo son:

- Configuración de parámetros del sistema.
- Creación y bloqueo de usuarios (solo médicos y laboratorios).

- Consulta de reclamos y recomendaciones ingresadas por los pacientes.
- Descarga de logs de errores.
- Visualizar / descargar auditoría.

1.4 Objetivo General

Implementar un sistema web de apoyo para el control y administración de la información generada por los médicos integrantes de la Sociedad de Trauma del Chimborazo.

1.5 Objetivos Específicos

- 1) Diseñar una herramienta que facilite al médico el análisis de las recomendaciones o diagnósticos realizados al paciente mediante información clínica importante relacionada al paciente.
- 2) Desarrollar una solución para obtener información inviolable de resultados de laboratorio, cuya veracidad brinde mayor confianza al médico para poder realizar diagnósticos más seguros.
- 3) Medir el nivel de satisfacción de los pacientes en relación con el servicio que le es brindado por los médicos y con los resultados obtenidos con las solicitudes que son realizadas por los mismos.

1.6 Metodología

En base al análisis de los estudios realizados en las diferentes observaciones en las actividades diarias realizadas y los requisitos solicitados por los integrantes de esta entidad, se optará por realizar un sistema web flexible y seguro, que va a ser implementado con una arquitectura Modelo-Web-Template utilizando el framework web Django, el cual es desarrollado en el lenguaje de programación Python.

Para brindar mayor seguridad a la información que se va a administrar mediante el sistema, se utilizará una base de datos Postgres, la misma que será alojada en otro servidor, el cual realizará respaldos diarios de dicha base.

Para el intercambio de información entre el sistema y la base de datos se implementará un sistema REST API alojado en otro servidor, este será implementado en el framework Django-REST. El sistema principal se comunicará con este sistema enviando una llave encriptada de autenticación, para así determinar si existe un acceso permitido a la información, brindando otra capa de seguridad al sistema global.

CAPÍTULO 2

MARCO TEÓRICO

2.1 Arquitectura Web Modelo-Vista-Template (MVT)

La arquitectura web MVT es basada en la arquitectura Modelo-Vista-Controlador (MVC).

Arquitectura Web Modelo-Vista-Controlador (MVC)

[1] Para poder entender la arquitectura MVT primero se necesita conocer el funcionamiento de la estructura MVC. Esta arquitectura consiste en separar los datos, la lógica de negocio de una aplicación de su

representación y el módulo encargado de gestionar los eventos y las comunicaciones. Esta arquitectura está definida por tres componentes, cuya interacción puede ser representada de la siguiente forma:

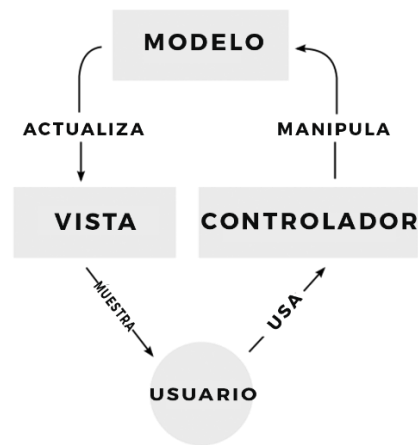


Figura 2.1: Funcionamiento de Infraestructura MVC

Fuente: El Autor

Modelo

Es la representación de la información, este componente gestiona todos los accesos y acciones que se puedan realizar a dicha información, es decir; ingreso, consulta, actualización y eliminación.

Controlador

Este componente responde a eventos o acciones solicitadas por el usuario o la aplicación, esto invoca peticiones al componente “modelo” en caso de que se solicite algún proceso o consulta a la información, para finalmente enviar esta información al componente “vista”; por lo que se podría decir

que este componente es el intermediario entre los componentes “modelo” y “vista”.

Vista

Este componente se encarga de representar a la información procesada por el componente “controlador”, de tal manera que dicha información pueda tener un formato con el que se pueda interactuar y siendo representada como salida, sin embargo, también se pueden hacer ligeros procesamientos de información en este componente antes de presentar la misma.

Estructura de Infraestructura Web Modelo-Vista-Template (MVT)

[2] Esta es una infraestructura que es utilizada exclusivamente por el Framework Django, el funcionamiento de la misma puede ser representada mediante la siguiente gráfica:

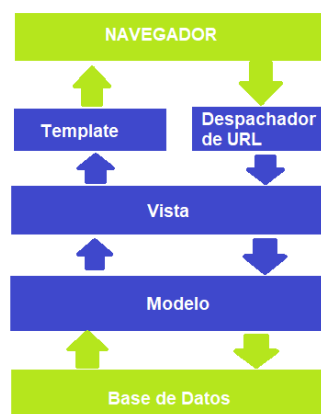


Figura 2.2: Funcionamiento de Infraestructura MVT

Fuente: El Autor

Esta infraestructura también consiste de 3 componentes: Modelo, Vista y Template; los cuales tienen las siguientes funciones:

Modelo

Este componente, al igual que en la infraestructura MVC, tiene la función de gestionar los diferentes procesos que se tienen que ejecutar en la información.

Vista

La “vista” es el componente que determina cual información es la que va a ser mostrada en la aplicación, esta información es obtenida mediante llamados al componente “modelo” para después enviarla al componente “template”. Cada tipo de página creada en la aplicación mediante el framework tendrá su respectiva “vista”, la cual es llamada mediante un interpretador de URL que es utilizado dentro de la infraestructura.

Este componente utiliza funciones ORM para poder gestionar las acciones a realizar en la información, esto permite que exista una mejora en el desempeño de la aplicación. También se pueden ejecutar otro de tipo de acciones como envío de correos, autenticaciones contra servicios externos, validaciones de datos enviados de formularios y modificación del contexto de la aplicación, en el que el contexto son variables disponibles en la aplicación que son enviadas al componente “template”.

Template

Las funciones de este componente es la de organizar los parámetros enviados en el contexto mediante la “vista”, utilizando también los atributos de los mismos, para poder ser mostrados en la aplicación; todo esto mediante un sistema de tags simples que se encargan de generar el código HTML. En este componente no se puede poner ningún código de otro lenguaje que no sea el de tags de manera arbitraria, tal como se puede realizar en otros modelos. Este sistema es muy limitado y solo tiene la funcionalidad de generar el código de presentación de información, lo cual brinda mayor seguridad en la estructura de la aplicación que se implemente mediante esta infraestructura.

Diferencia entre Infraestructura MVC y MVT

Ambas infraestructuras difieren en cómo son ejecutadas los diferentes componentes de cada una. En el caso de la infraestructura MVC, es utilizado el cambio de estados para poder mantener todo sincronizado; todos los componentes existen en memoria al mismo tiempo, posiblemente ejecutándose en diferentes hilos o procesos, por largos períodos de tiempo, teniendo cada uno su propio estado y teniendo interacciones entre cada uno.

La infraestructura MVT no utiliza estados, solo usa los datos, es decir, la información que es obtenida de la base de datos es tratada como

información inmutable en el componente “template”, el cual únicamente presenta la misma, a diferencia de la infraestructura MVC que tiene que utilizar memoria para poder procesar la información a mostrar en su componente de presentación, la “vista”.

2.2 Arquitectura de acceso de datos REST API

La arquitectura REST API es una combinación de dos metodologías de programación, la que es utilizada comúnmente para la implementación de web services, la cual es una forma para obtener o procesar información.

API es una Interfaz de Programación de Aplicaciones en la cual se determinan reglas para permitir la comunicación entre varias aplicaciones mediante métodos, definidos por el usuario, que contienen la lógica para procesar o consultar información.

[3] REST determina como la API se va a ver o como va a ser accedida, REST significa Transferencia de Estados Representacionales y son un conjunto de reglas que los desarrolladores deben seguir para crear una API; estas reglas sirven para poder tener acceso a los métodos que procesan o consultan la información determinados en la API que devuelve la información requerida, esta puede ser accedida mediante la respuesta obtenida a llamados a un URL específico cuya estructura es determinada y procesada mediante uno de los componentes de la aplicación REST, en el cual también se encuentran los parámetros necesarios para procesar la

información dentro de su estructura; el formato de respuesta es determinado también por la misma aplicación. Entre los formatos más utilizados están XML y JSON, ya que por la estructura de información que brindan son fáciles de interpretar.

Estructura de REST

Los elementos estructurales de un REST son: conectores, componentes y elementos de datos.

Conectores

Los conectores representan las actividades involucradas en el acceso de recursos y transferencia de representaciones de la información. REST encapsula las actividades existentes en diferentes tipos de conectores. En la siguiente tabla se muestran los tipos de conectores existentes:

Tabla 1: Tipos de Conectores en Estructura de REST

Tipo de Conector	Descripción	Ejemplo
Cliente	Envía peticiones, recibe respuestas.	Librería HTTP
Servidor	Escucha peticiones, envía respuestas.	Servidor de Web API

Tipo de Conector	Descripción	Ejemplo
Caché	Puede estar ubicada en el lado del cliente o en el servidor conector para almacenar respuestas que puedan estar en memoria.	Caché del navegador
Despachador	Transforma los identificadores de recursos en direcciones de red.	Bind (Librería de búsqueda del DNS)
Túnel	Pasa las peticiones generadas, cualquier componente puede cambiar de comportamiento activo a comportamiento de túnel.	SOCKS y SSL después de la conexión HTTP

Componentes

Estos son los diferentes tipos de software que se comunican entre sí. Estos están categorizados por roles, los cuales son los siguientes:

Tabla 2: Roles de Componentes de Estructura REST

Rol de Componente	Descripción	Ejemplo
Servidor de Origen	Usa un servidor conector para recibir peticiones, y es la fuente definitiva para la representación de sus recursos mediante una interfaz.	Apache httpd y Glassfish
User Agent	Usa al cliente para iniciar peticiones y también el receptor final de las respuestas.	Navegadores de internet como Firefox, Chrome y Opera
Gateway	Actúa como ambos, cliente y servidor para poder enviar peticiones y respuestas.	Squid y CGI
Proxy	Hace de intermediario entre las peticiones que realiza el servidor cliente a otros servidores.	Proxy de Windows

Elementos de Datos

El principal aspecto de REST es el estado de sus elementos de datos, sus componentes se comunican por la transferencia de representaciones de los estados actuales o deseados de los elementos de los datos. REST identifica 6 elementos de datos, los cuales son los siguientes:

Tabla 3: Elementos de Datos de Estructura REST

Elemento de Datos	Descripción	Ejemplo
Recurso	Cualquier información puede ser llamada como recurso. Un recurso es un mapeo conceptual a un conjunto de entidades, no una entidad en sí. Este mapeo puede cambiar con el tiempo. En general, un recurso REST es cualquier cosa que pueda ser direccionable en la web.	El título de una película, un video de Youtube o una imagen
Identificador de recurso	Cada recurso debe tener un nombre único que lo identifique. Esto es llamado URI (Uniform Resource Identifier) bajo HTTP. URI en un sistema REST, es un	<p>Formato estandarizado de URI:</p> <p>esquema://host:puerto/ruta?queryString</p> <p>#fragmento.g:</p> <p>http://dominio.com/informacion?id=123</p>

	<p>hipervínculo al recurso.</p> <p>Esta es la única forma para cliente y servidores de intercambiar representaciones de recursos. La relación entre URIs y el recurso es muchos a uno. Un recurso puede tener varios URIs que provee diferente ubicación acerca de la ubicación del recurso.</p>	
<p>Metadata de recurso</p>	<p>Esto describe el recurso. Una metadata provee información adicional como la ubicación de la información.</p>	<p>Link de origen</p>
<p>Representación</p>	<p>Es algo que es enviado, ida y vuelta, entre los clientes y los servidores. Entonces, nunca se envían o reciben recursos, solo sus representaciones. Una</p>	<p>Secuencia de bytes, documento HTML o documento de imagen</p>

	representación captura el estado actual o destinado del recurso. Un recurso puede tener varias representaciones.	
Metadata de Representación	Describe la representación.	Cabeceras (tipo de medio)
Datos de control	Esto define el propósito de un mensaje entre componentes o la acción que haya sido solicitada.	If-Modified-Since, If-Match

Que sea direccionable significa que pueda ser accedido y transferido entre clientes y servidores.

Los detalles del formato estandarizado de un URI son los siguientes:

esquema://host:pueto/ruta?queryString#fragmento

- Esquema: Es el protocolo que se está utilizando para comunicarse. Para un REST web service, normalmente se usa http o https.
- Host: Es el nombre del DNS o dirección IP.
- Puerto: Esto es opcional y tiene que ser numérico. El host y el puerto representan la ubicación del recurso en la red.

- Ruta: Esta expresión es un grupo de segmentos de texto delimitados por el carácter “/”. La expresión de la ruta está pensada como una lista de directorios de un archivo en una máquina.
- ?: Este carácter es el separador entre la ruta y el queryString.
- QueryString: Es una lista de parámetros representados como pares de nombre/valor. Cada par es separado por el carácter delimitador “&”.
- Fragmento: Es delimitado por el carácter “#”. El fragmento es normalmente utilizado para ubicar un punto de un lugar exacto de un documento que se esté consultando.

Restricciones de REST

[4] Estas son reglas de diseño que son aplicadas a las diferentes características de una arquitectura REST. Si se cumplen todas las restricciones de diseño, entonces se podría considerar que un sistema es RESTful.

Estas restricciones definen como los datos son transferidos entre componentes y los beneficios que brindan, sin depender de la tecnología que se esté utilizando o la arquitectura que se haya implementado en el sistema.

Estas restricciones son:

- Cliente-Servidor.
- Stateless.
- Cacheable.
- Interfaz Uniforme.
- Sistema por capas.
- Código en demanda (opcional).

Cliente-Servidor

Se requiere la existencia de componente cliente que envíe pedidos y un componente servidor que reciba el pedido, para que el componente servidor envíe una respuesta que sea recibida por el componente cliente.

Stateless

El servidor no puede recordar ningún estado de la aplicación, esto quiere decir, que el lado del cliente debe enviar toda la información necesaria para la ejecución solicitada en el pedido al servidor, ya que el servidor no podrá reutilizar la información de un pedido anterior y tampoco recordará dicha información.

Cacheable

En caso de que la información ya se encuentre almacenada en el caché de la aplicación REST, entonces ya no será necesario realizar el pedido al servidor, ya que esta información puede ser reutilizada o parcialmente reutilizada, dependiendo de la configuración que se haya realizado en la aplicación REST.

Interfaz Uniforme

Esta restricción a un alto nivel significa que la interfaz para un componente debe ser lo más genérica posible. Esto simplifica y desacopla la arquitectura, lo que permitiría que cada parte de la arquitectura evolucione de manera independiente.

Sistema por Capas

Pueden existir capas entre el cliente y el servidor. Estas pueden ser llamadas intermediarias y pueden ser agregadas en varios puntos entre el pedido y la respuesta sin cambiar la interfaz entre los componentes, estos pasan mensajes tales como traducciones o improvisar el desempeño con el caché. Los intermediarios pueden ser los proxies o los gateways, en lo que los proxies son escogidos por el cliente y los gateways son escogidos por el servidor de origen o impuestos por la red.

Código en demanda

Permite a un cliente descargar y ejecutar el código de un servidor. Esto es opcional.

Ventajas del uso de REST API

Una ventaja que brinda el uso de REST API es que puede brindar acceso a información a todo tipo de aplicaciones o programas, sin importar la plataforma o lenguaje de programación en el que este se haya implementado, ya que únicamente se necesita hacer llamados a determinados URLs para poder obtener y/o procesar información. Por otra parte esto ayuda al desempeño del servidor en el que se encuentra ejecutándose la aplicación, ya que separa la parte de la lógica de la aplicación con la lógica que procesa y consulta la información que se va a mostrar, ya que esta lógica sería ejecutada mediante la aplicación REST.

2.3 Seguridad de la información mediante REST API y web tokens

[5] Los web tokens son un conjunto de protocolos de identificación abiertos que utilizan representaciones de datos JSON y sistemas de comunicación REST. Estos protocolos fueron diseñados para que fueran fácil de utilizar por navegadores web y por ambientes modernos de desarrollo web, los cuales típicamente tienen soporte de JSON.

En este caso se implementará el uso JSON Web Token (JWT) como medida de seguridad.

JWT representa un conjunto de información como un objeto JSON que es codificado con una estructura de Firma Web JSON (JWS) y/o Encriptación Web JSON (JWE). El objeto JSON consiste en cero o más nombres/valores apareados (o miembros), donde los nombres son strings y los valores arbitrarios JSON. Los nombres de los miembros son llamados Claim names y los valores son llamados Claim Values.

En el contenido del JWT se encuentra la cabecera JOSE (JSON Object Signing and Encryption) que describe la operación criptográfica que se va a aplicar en la información del JWT. Si la cabecera JOSE se va usar para un JWS, el JWT es representado como un JWS y la información es firmada digitalmente o MACed, con la información del JWT siendo cargada en el JWS. Si la cabecera JOSE se va a usar en un JWE, el JWT es representado como un JWE y la información es encriptada, siendo la información del JWT un texto plano encriptado por el JWE.

El JWT es representado como una secuencia de partes de una URL segura separadas por puntos. Cada parte contiene valores de URL codificados en base64. Las partes en un JWT depende de la representación resultante del JWS usando la serialización compacta del

JWS o la representación del JWE resultante del uso de la serialización compacta del JWE.

[6] Por ejemplo, la siguiente cabecera JOSE:

```
{"typ":"JWT",  
  
  "alg":"HS256"}
```

Declara que el objeto codificado es un JWT, el cual es un JWS que va a ser firmado digitalmente usando el algoritmo HMAC SHA-256. Para remover ambigüedades potenciales se convierte a la cabecera JOSE en una representación UTF-8 basada en octetos para luego codificarlo en base64, teniendo como resultado lo siguiente:

```
eyJ0eXAiOiJKV1QiLA0KICJhbGciOiJIUzI1NiJ9
```

Después se realiza el mismo proceso en la información del JWT, por ejemplo, la siguiente información:

```
{"iss":"joe",  
  
  "exp":1300819380,  
  
  "http://example.com/is_root":true}
```

Se la convierte en una representación UTF-8 basada en octetos para luego codificarla en base64, teniendo como resultado lo siguiente:

eyJpc3MiOiJqb2UiLA0KICJleHAiOjEzMDA4MTkzODAsDQogImh0dHA6Ly9leGFtcGxlIjMvS9pc19yb290Ijp0cnVifQ

Siendo esto cargado en el JWS, después de obtener la firma digital mediante un cálculo utilizando el algoritmo de encriptación HMAC SHA-256 entre la cabecera JOSE y el JWS y luego convertirlo en base64. Como resultado se tendría lo siguiente como firma digital:

dBjftJeZ4CVP-mB92K27uhbUJU1p1r_wW1gFWFOEjXk

Finalmente se concatenan los resultados en base64 de la cabecera JOSE, el JWS y la firma digital mediante puntos. Resultando lo siguiente:

eyJ0eXAiOiJKV1QiLA0KICJhbGciOiJIUzI1NiJ9.

eyJpc3MiOiJqb2UiLA0KICJleHAiOjEzMDA4MTkzODAsDQogImh0dHA6Ly9leGFtcGxlIjMvS9pc19yb290Ijp0cnVifQ.

dBjftJeZ4CVP-mB92K27uhbUJU1p1r_wW1gFWFOEjXk

Siendo esto el JWT encriptado que se utilizaría para autenticar en el sistema y que sería enviado mediante la cabecera de pedido al sistema REST.

A este proceso se lo puede representar mediante la siguiente gráfica:

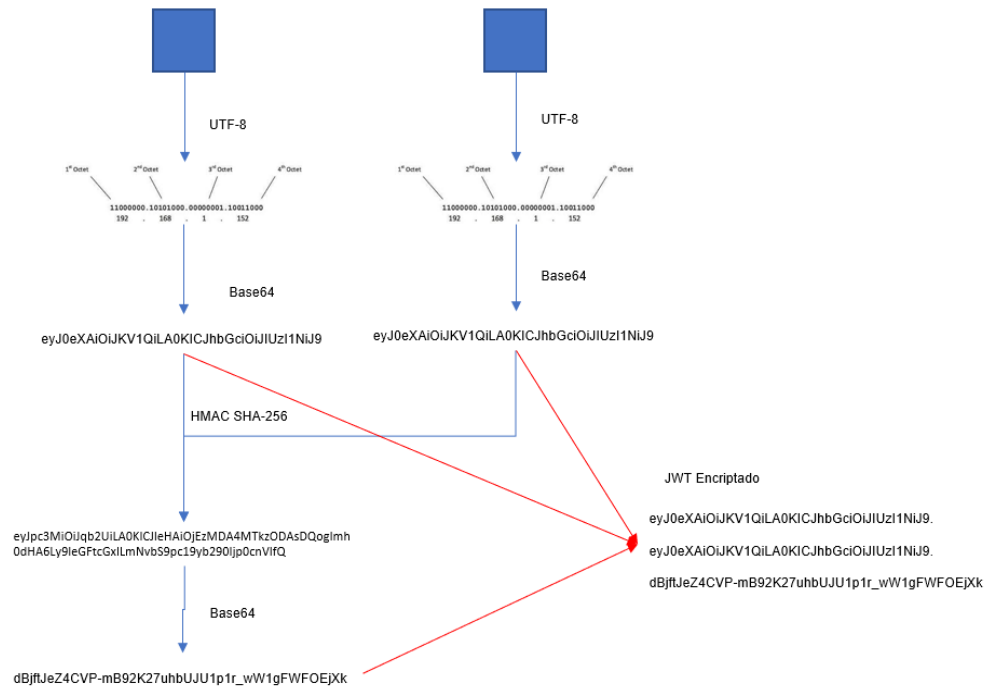


Figura 2.3: Estructura de JWT Encriptado

Fuente: El Autor

2.4 Normativa de calidad ISO 13485

[7] La normativa ISO 13485 representa a todos los requisitos para un sistema de gestión, para el diseño y fabricación de dispositivos médicos, ya sea en dispositivos o servicios relacionados, esto para asegurar que los requisitos de los clientes y normativa se cumplan de forma consistente.

Esta norma está basada en la norma ISO 9001, la cual consiste en un conjunto de criterios que conforman los estándares de un sistema administrativo de calidad. Estos estándares están basados en principios administrativos de calidad incluyendo un énfasis en la atención al cliente,

motivación, una alta implicación en la administración, enfoque en los procesos y mejoras continuas.

La norma 13485 cubre regulaciones para compañías involucradas en cualquier escenario que involucre al diseño, desarrollo, producción, almacenamiento y distribución de equipos médicos y servicios.

Este estándar se enfoca principalmente en los enfoques basados en riesgos, la documentación de habilidades y entrenamientos, administración de quejas, monitoreo de proveedores, diseño y desarrollo.

Para poder obtener la certificación de esta norma se deben realizar los siguientes pasos:

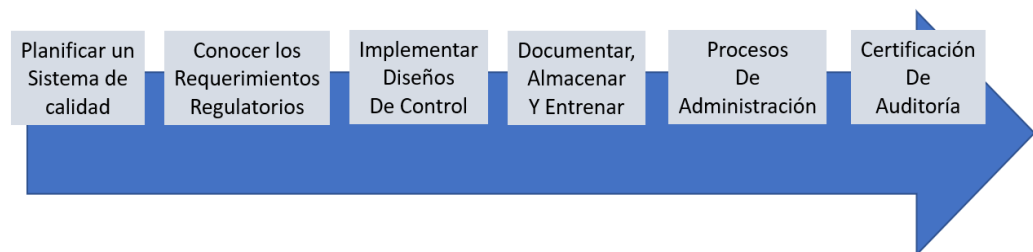


Figura 2.4: Procesos de Norma ISO 13485

Fuente: El Autor

Planificar el Sistema de Calidad

Es necesario documentar los planes de calidad que se deseen implementar en el sistema. Esto no requiere ningún formato específico, generalmente se utilizan diagramas de Gantt.

En el plan de calidad se deben identificar todos los procedimientos requeridos para el sistema de calidad, las secuencias de ejecución e implementación de los documentos generados. También debe existir un manual de calidad. En el plan debe constar el manual, los procedimientos de control de documentos y almacenamiento, y la seguridad de tener los recursos necesarios para implementar el mismo. Los recursos pueden ser personas, dinero y tiempo.

Conocer los requerimientos regulatorios

Se debe definir los mercados globales que van a poder legislar en el plan de calidad. En este caso, el mercado es la Sociedad de Trauma, Capítulo Chimborazo.

Implementar diseños de control

Este es uno de los puntos más difíciles que deben dominar las compañías que implementan esta ISO bajo contrato con otras compañías; esto debido a que se tiene de revisar detenidamente los niveles de control que van a tener los usuarios dentro del plan.

Documentación, almacenamiento y entrenamiento

Este es el proceso más importante de esta regulación, ya que permite dar seguimiento a los cambios que se hayan realizado. Una vez que se hayan aprobado los procesos de documentación, almacenamiento y diseño, se

debe realizar la documentación para entrenamiento. La manera de documentar el entrenamiento es muy importante; se debe registrar el entrenamiento, la efectividad del entrenamiento y la competencia.

Procesos de Administración

Los procesos de administración principales son: CAPA (Corrección, Acción, Prevención y Acción), auditoría interna y revisión administrativa, siendo estos provistos por el sistema de calidad. El propósito de la auditoría interna es identificar los problemas en el sistema de calidad. El proceso CAPA identifica la fuente de esos problemas y los elimina. La revisión administrativa se asegura de que los recursos adecuados estén disponibles para mantener la efectividad del sistema de calidad.

Certificación de Auditoría

La normativa ISO 17021 es un estándar que define los requerimientos de aquellos que proveen auditoría de la ISO 13485 y certificaciones en sistemas de administración. Estos requerimientos incluyen dos escenarios. El primer escenario consiste en la revisión de la documentación del sistema de calidad y verificar que la empresa esté preparada para el segundo escenario; que consiste en una revisión profunda de los procesos. Actualmente en estos escenarios también se incluyen revisiones de registros de la auditoría interna, CAPA y administración de revisiones de procesos.

Requerimientos

Los documentos requeridos para el desarrollo, implementación y mantenimiento de un sistema de calidad bajo esta normativa son los siguientes:

Tabla 4: Requerimientos de Normativa ISO 13485

Requerimiento	Descripción
Manual de calidad	Ayuda en la administración del sistema de calidad. Incluye el objetivo del sistema, los roles existentes en la empresa, exclusiones de estándares, referencias a documentos relevantes y el modelo de negocio del proceso.
Procedimientos del Sistema de Calidad	Son necesarios para una efectiva planificación, operación, control y monitoreo de la ejecución de las mejoras en los procesos. Esto es de mucha utilidad al momento de realizar mantenimientos.
Exhibiciones	Son muy útiles para cubrir todos los detalles para los entrenamientos del

Requerimiento	Descripción
	usuario para implementar procesos y tener ideas para mejorar los mismos.
Formatos	Están diseñados y requeridos para mantener registros y establecer control, estas son guías que pueden cambiar de acuerdo a los requerimientos que vayan surgiendo.
Procedimientos Estándar de Operación (SOP)	Estos procedimientos toman una muestra de las instrucciones de trabajo para vincularlos con aspectos significativos de los problemas de la empresa. Son usados como una guía de entrenamiento y para controlar y hacer sistemas en la empresa.
Lista de Auditoría	Los ítems de la lista están basados en preguntas realizadas por los departamentos de la empresa. Esto es una buena herramienta para que los auditores hagan cuestionarios o cláusulas

Requerimiento	Descripción
	inteligentes para poder realizar auditorías de manera efectiva.

Atributos

El principal objetivo del ISO 13485 es armonizar los requerimientos regulatorios para los sistemas de calidad de los servicios médicos. Mediante la implementación de esta normativa en un sistema se puede obtener los siguientes atributos como ventajas:

- Acceso a un mercado más amplio, debido a la demanda de las aplicaciones de administración de calidad en el sector médico. El invertir en un certificado de calidad ayuda a mejorar la confianza con el cliente y eso al mismo tiempo mejora las posibilidades de que otros países tomen en cuenta al uso de la aplicación, por lo que acelera el acceso al mercado.
- Reducir costos de ventas. Al certificar una aplicación se demuestra de manera más fácil la efectividad del sistema de calidad y requiere menos tiempo en ganar la confianza de futuros clientes.
- Mejorar el desempeño de la aplicación. Al aplicar los estándares de

esta normativa al momento de desarrollar una aplicación, la estructura de la misma ya estará bajo los controles necesarios para mejorar sus procesos internos.

Procedimientos de Requerimientos

Los procedimientos de requerimientos son la clave principal para poder organizar y completar la documentación para crear un sistema de calidad y para desarrollar mejoras de manera más rápida.

Los requerimientos de procedimientos son los siguientes:

- Procedimiento para revisión de administración.
- Procedimiento para control de documentos y datos.
- Procedimiento para control de registros.
- Procedimiento para auditoría interna.
- Procedimiento para entrenamiento.
- Procedimiento para acciones correctivas y preventivas.
- Procedimiento para control de monitoreo y medida de equipos
- Procedimiento para control de monitoreo de ambiente de trabajo.
- Procedimiento para validación para proceso de esterilización.

- Procedimiento para monitoreo y medida de procesos.
- Procedimiento para análisis de datos
- Procedimiento para problemas e implementación de notificaciones de advertencia.
- Procedimiento para encuestar la satisfacción del cliente.
- Procedimiento para control de productos ajenos.
- Procedimiento para compras.
- Procedimiento para identificación de productos.
- Procedimiento de seguimiento.

CAPÍTULO 3

SITUACIÓN ACTUAL Y REQUERIMIENTOS

3.1 Situación actual

La Sociedad de Trauma del Ecuador – Capítulo Chimborazo, es una asociación que forma parte de La Sociedad de Trauma del Ecuador. En Chimborazo, tiene como miembros actuales a 25 médicos de diferentes especialidades, entre las que están fisioterapia, neurología, medicina general, pediatría, cirugía y endocrinología. Cada miembro cuenta con un consultorio médico personal en el cual se trata las enfermedades de sus distintos pacientes, teniendo pacientes en común, y cuya información

médica es de interés entre todos los miembros, ya sea, tanto para aplicar medicamentos o tratamientos, como para poder tener datos que sean útiles para poder generar futuras investigaciones en conjunto relacionadas a la medicina.

Cada miembro cuenta con diferentes métodos para poder administrar la información que generan de manera individual, entre estos métodos se encuentran software que ya son muy antiguos y que no tienen compatibilidad con los sistemas operativos actuales; contienen errores internos, pero no existe soporte técnico disponible o inclusive ya se encuentran descontinuados; utilizan software de apuntes para tener estos registros, como por ejemplo bloques de notas, Microsoft Word o Microsoft Excel; utilizan software propios desarrollados por terceros, los cuales una vez se han finalizado el desarrollo del mismo, se pierde el soporte técnico, sin dejar la oportunidad de desarrollo de nuevas funcionalidades o correcciones de errores que se han encontrado con el pasar del tiempo; no utilizan software alguno y mantienen los registros en cuadernos físicos; o como último no mantienen registro alguno de su información. Cabe recalcar que ninguno de estos métodos permite vincular la información de los miembros.

Existe una molestia de ocurre de manera recurrente, la cual es la solicitud de información médica por parte de los pacientes de los médicos

integrantes de esta sociedad; esto ocurre debido a que la solicitud hay que realizarla de manera personal y en muchas ocasiones los médicos no se encuentran presentes o disponibles, por lo que no hay una manera rápida de obtener esa información.

Actualmente, como no existe un vínculo entre la información generada por los médicos integrantes de la Sociedad de Trauma, entonces existen varios historiales médicos de los pacientes, cada uno diferente por médico, por lo que para poder obtener un historial médico completo y detallado, existe la necesidad de realizar múltiples solicitudes a los diferentes médicos que han atendido al paciente, volviendo a producirse el problema mencionado en el párrafo anterior.

En la actualidad, no existe ninguna forma de poder obtener retroalimentación por parte de los pacientes acerca del servicio brindado.

Por último, también han existido casos de resultados de laboratorio alterados y que han sido entregados a los médicos en las consultas por parte de los pacientes, por lo que no existe seguridad de la información que es solicitada por parte del médico.

3.2 Requerimientos funcionales

La Sociedad de Trauma ha pedido como solución un sistema web, el cual pueda ser utilizado por todos los miembros de la sociedad, de tal manera

que la información sea generada únicamente desde este sistema, para poder obtener un vínculo de información entre los miembros. Esta debe tener una interfaz intuitiva y capaz de poder ser utilizada desde cualquier dispositivo, siendo estos computadores, celulares y tablets.

El sistema podrá generar usuarios de nuevos miembros de la sociedad por un usuario administrador técnico. Este usuario administrador también podrá realizar cambios en la configuración del sistema en caso para acoplar su funcionalidad dependiendo de los requerimientos que vayan surgiendo con el tiempo. El administrador también podrá generar perfiles de usuario que permitirán administrar los permisos a acciones que podrán realizar los usuarios que se encuentren en un perfil específico dentro del sistema, por ejemplo, si un miembro tiene un asistente que solo pueda ingresar información al sistema, mas no modificarla, entonces se creará un perfil que tenga permisos solo para esas funciones.

En el sistema, los miembros de la sociedad podrán registrar, modificar y eliminar, de manera lógica, la información generada en su día laboral; siendo esta, los datos personales de los pacientes, las consultas médicas realizadas a los mismos, los medicamentos recetados, los datos médicos del paciente, siendo estos: estatura, peso, edad, niveles de glucosa, operaciones realizadas, alergias, entre otros datos médicos de interés que pueden ser configurados por el propio miembro, y los cuales almacenarán

el historial de estos datos en caso de que se realicen cambios; como requisito obligatorio se debe también generar de manera automática el BMI (Body Mass Index) del paciente una vez que se hayan ingresado los datos necesarios para calcularlo, se registrará el historial del mismo, este dato es un indicador que es muy utilizado por todos los médicos para poder tener un diagnóstico inicial del estado del paciente. Esto se realizará con el objetivo de generar un historial médico detallado del paciente y reportes de los avances obtenidos del paciente, analizar sugerencias a realizar al paciente en base a los reportes para evitar futuras posibles enfermedades.

La información generada del paciente por cada miembro de la sociedad podrá ser compartida a otros miembros y generar más información por los mismos, solo en caso de que dicho paciente tenga que ser revisado por otro de los miembros.

El paciente podrá acceder al sistema, de manera voluntaria e inmediata, para poder consultar la información que ha sido registrada. El paciente podrá ingresar retroalimentación de sus consultas para poder medir el nivel de calidad de la atención al mismo, esta información será analizada por personal de auditoría interna para recomendar futuras acciones a realizar para mejorar la atención y consecuente satisfacción por parte del paciente.

El sistema podrá ser accedido por laboratorios médicos vinculados con la sociedad. El personal que trabaja en estos laboratorios podrá ingresar de

manera directa los resultados de los exámenes médicos de pacientes solicitados por los miembros de la sociedad, para poder ser analizados de manera inmediata por dichos miembros. Estos podrán consultar la información de los exámenes que hayan sido realizados anteriormente. Esto brindará mayor seguridad y confianza en la información generada de los resultados obtenidos y un respaldo de los mismos.

El sistema debe ser expansible y flexible, es decir, futuros desarrolladores podrán realizar nuevos módulos con nuevas funcionalidades dependiendo de los requerimientos tanto del médico como de los pacientes, sin la necesidad de realizar cambios fuertes e innecesarios dentro de la aplicación.

Se debe realizar respaldos de la información generada de manera periódica para no perder datos en caso de algún fallo inesperado del ambiente en el que se está ejecutando el sistema.

3.3 Procesos a optimizar

Los procesos diarios realizados por los miembros de la sociedad y que se van optimizar son los siguientes:

Ingreso y actualización de información médica de los pacientes

Los miembros podrán ingresar de manera más rápida, segura y eficiente la información que se vaya generando de los pacientes diagnosticados

mediante citas médicas. También podrá actualizar los datos médicos del paciente en caso de que sea necesario.

Consulta de historial médico del paciente

Toda información ingresada o actualizada por los miembros de la Sociedad de Trauma acerca de un paciente formará de manera automática parte del historial médico del mismo, el cual podrá ser consultado y visualizado de manera detallada por parte de los miembros que hayan atendido a dicho paciente.

Análisis de datos médicos del paciente y diagnósticos a realizar

El sistema mostrará directamente los datos médicos del paciente como indicadores, que hayan sido ingresados por los miembros de la sociedad. También se mostrará gráficas que muestren las variaciones de los valores de estos indicadores, esto ayudará a agilizar el análisis de esta información, que permitirá realizar un mejor diagnóstico del paciente. Por otra parte, también podrá acceder y exportar reportes generados con la información ingresada.

Obtención de información del paciente tratado por otro miembro

El sistema podrá brindar acceso directamente a la información médica de los pacientes generada por otros miembros, en caso de que realice un diagnóstico de dichos pacientes en alguna consulta médica.

Análisis de acciones a realizar para mejorar la satisfacción de los pacientes

Mediante la información obtenida por la retroalimentación de los pacientes, y el análisis de la misma, se pueden plantear acciones a realizar con los miembros de la sociedad para poder mejorar el nivel de satisfacción del servicio brindado a los pacientes.

Obtención de resultados en exámenes médicos realizados en laboratorios

La solicitud y obtención de los resultados de exámenes de laboratorio podrán ser realizados desde el sistema, al igual la información de dichos resultados serán ingresados por el personal que trabaja en el laboratorio vinculado de manera directa mediante el sistema; por lo que la información de estos resultados no puede ser manipulada por parte del paciente, siendo estos seguros y confiables. Adicionalmente se disminuye el tiempo de ejecución de todo el procedimiento.

Acceso a la información médica personal al paciente

Mediante el ingreso al sistema por parte del paciente, este podrá acceder a toda su información médica generada por los miembros de la sociedad, tal como, información personal, datos médicos como estatura, peso, edad, niveles de glucosa, operaciones realizadas, alergias, BMI, entre otros;

consultas médicas realizadas con los diferentes médicos miembros de la Sociedad de Trauma junto con sus respectivos diagnósticos, tratamientos, sugerencias, futuras citas médicas, retroalimentaciones realizadas y las respuestas a las mismas.

Obtención y análisis de información para estudios e investigaciones médicas

Con la información ingresada por los miembros de la Sociedad de Trauma, el sistema realizará un análisis de esta información de manera anónima, es decir, sin tomar en cuenta los datos personales del paciente, únicamente utilizando los datos médicos; para poder obtener resultados estadísticos o médicos que puedan ser de utilidad para iniciar o soportar estudios e investigaciones realizadas de manera conjunta o individual por parte de los miembros de la Sociedad de Trauma.

3.4 Actores y casos de flujo

Actores

Los actores que se pueden identificar son los siguientes:

- Médico.
- Paciente.
- Laboratorista.

- Asistente médico.
- Administrador o Técnico del Sistema.
- Auditor Interno.

Casos de Uso

Como se mencionó en la solución propuesta, el sistema va a estar compuesto en los siguientes módulos: Médico, Paciente, Laboratorio y Sistemas. De los cuales se pueden obtener los siguientes casos de uso.

Módulo Paciente

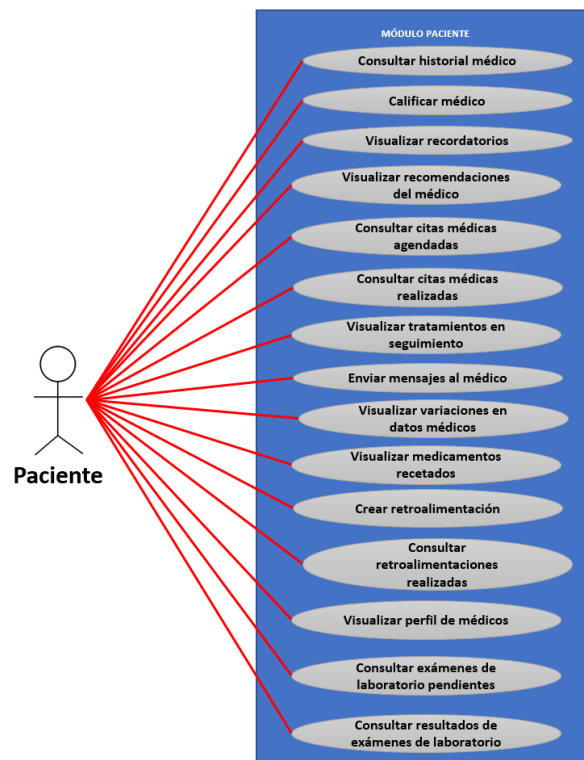


Figura 3.1: Caso de Uso - Módulo Paciente

Fuente: El Autor

Módulo Médico

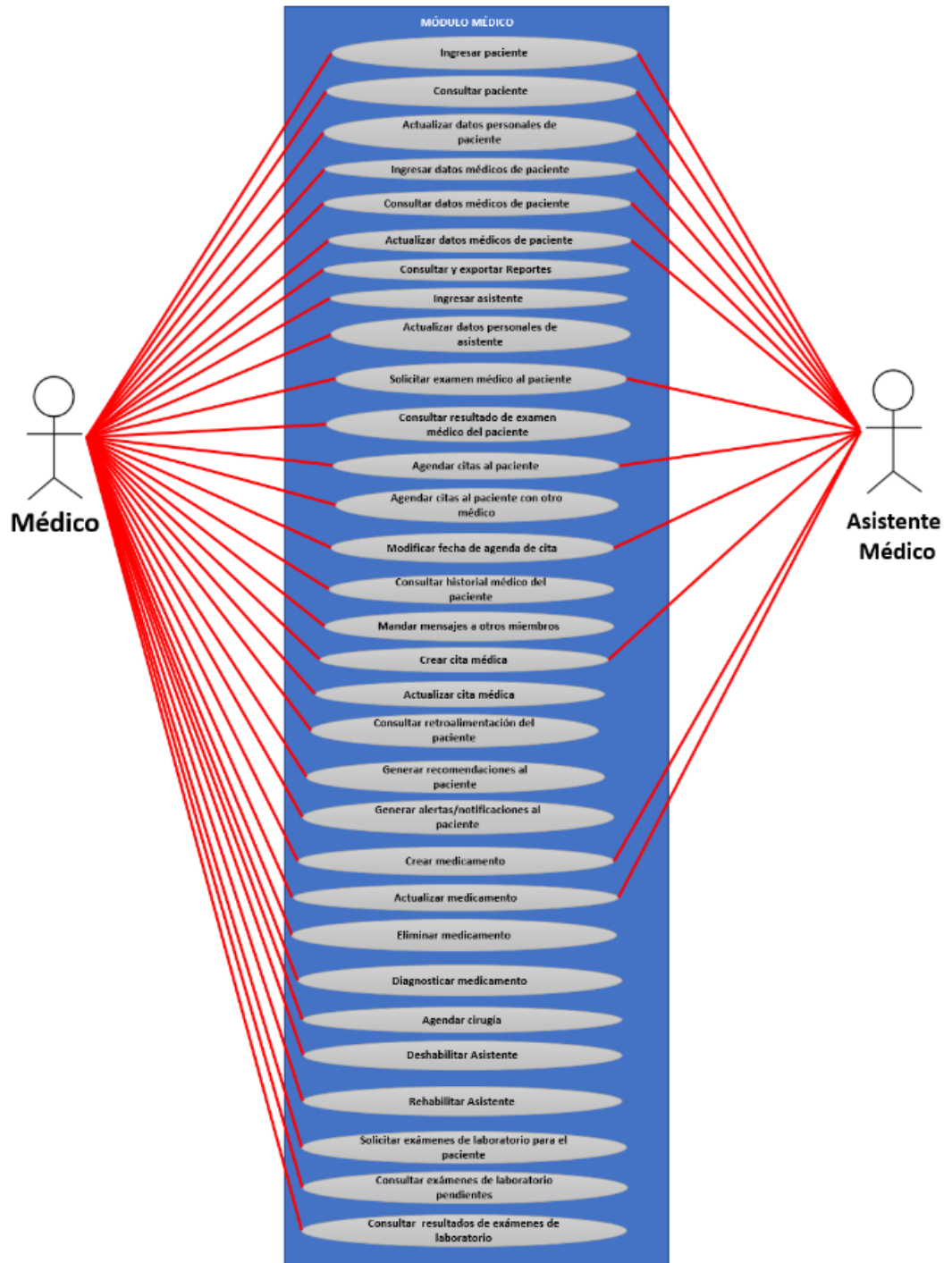


Figura 3.2: Caso de uso - Módulo Médico

Fuente: El Autor

Módulo Laboratorio

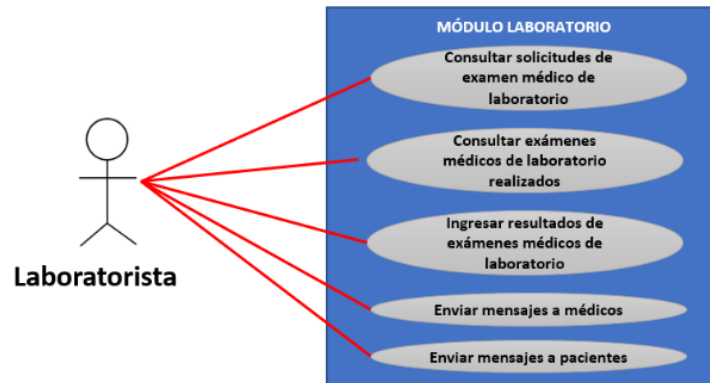


Figura 3.3: Caso de Uso - Módulo Laboratorio

Fuente: El Autor

Módulo Sistemas

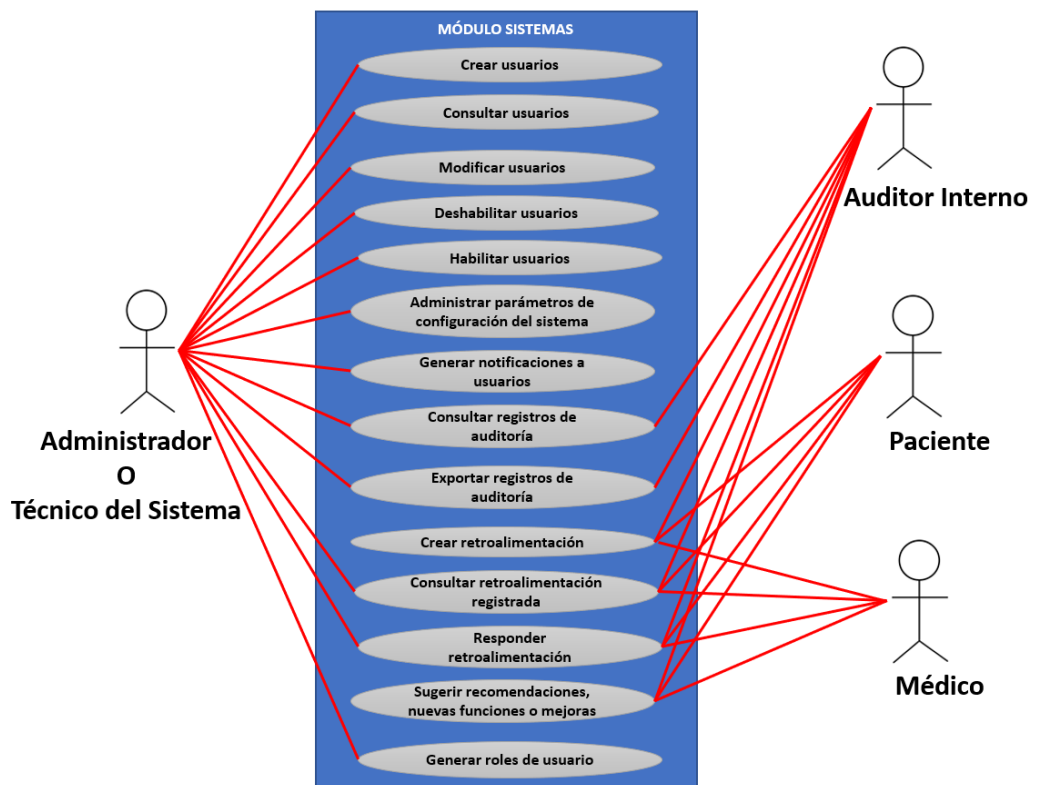


Figura 3.4: Caso de Uso - Módulo Sistemas

Fuente: El Autor

CAPÍTULO 4

ANÁLISIS Y DISEÑO

4.1 Análisis de Web Services a implementar

Como se mencionó en la solución propuesta, una parte del sistema global a implementar, es un sistema REST API que permitirá la comunicación entre el componente de interfaz del sistema global y la base datos; también será el encargado de procesar los parámetros tipo POST que son enviados desde la interfaz para poder ejecutar los procesos adecuados que retornará la información o realizarán las acciones que fueron solicitadas; verificando primero si el usuario que realiza la petición de información tiene

privilegios para acceder a la misma mediante el uso de web tokens encriptados que son enviados en la cabecera de la petición HTTP.

Para que funcione correctamente el sistema, es necesario definir las posibles acciones que se podrán ejecutar o el tipo de información que se podrá solicitar mediante el sistema REST API, para poder definir web services necesarios que serán implementados en dicho sistema.

Para tener acceso al sistema REST API, se realizarán las peticiones a “http://dominio:puerto/”, donde, “dominio” va a ser el dominio final en el que se van a acceder a los recursos del sistema REST API y el “puerto” es el puerto dedicado del dominio que estará destinado para poder acceder al mismo sistema, indicando que esta información aún no está definida.

Para poder hacer más simple la comprensión de la estructura de las URIs completas que se van a implementar en el sistema, se ha diseñado un formato basado en términos reservados definidos en el diseño de la estructura del sistema, los cuales describen la acción y el tipo de objeto que estará involucrado en el proceso que se va a ejecutar. Por ejemplo, el esquema descriptivo de este tipo de URIs puede ser el siguiente:

http://dominio:puerto/término-de-sistema/término-de-módulo/término-de-tipo-de-acción/término-de-objeto/término-de-acción

Los parámetros enviados al sistema REST API son de tipo POST, por lo que no tienen que ser declarados con un QueryString. Los términos reservados son los siguientes:

Términos de Sistema

El único término definido de sistema es "sistema". En caso de que el sistema se amplíe y se cree algún subsistema, entonces se determinará algún nuevo término para que defina dicho subsistema.

Términos de Módulos

Este término indica el módulo en el que se encuentra el usuario en el sistema. Estos términos son los siguientes:

Tabla 5: Términos reservados para Módulos

Término	Descripción
Médico	El usuario se encuentra actualmente en el módulo Médico.
Paciente	El usuario se encuentra actualmente en el módulo Paciente.
Laboratorio	El usuario se encuentra actualmente en el módulo Laboratorio.

Término	Descripción
Sistemas	El usuario se encuentra actualmente en el módulo Sistemas.

Términos de tipo de acción

Este término define el tipo de acción que se va a ejecutar al momento de realizar peticiones a una URI definida en el sistema REST API, estos términos son exactos y no se puede utilizar ningún otro término que no conste en los predefinidos en el sistema. Los tipos de acciones especificadas son las siguientes:

Tabla 6: Términos reservados de tipos de acción

Término	Descripción
cons	Define a una consulta de información, determinando también que la respuesta es un conjunto de objetos de información.
proc	Define al procesamiento de información, esto puede ser inserción, actualización y eliminación de datos. La respuesta obtenida puede ser 1, en caso de que no haya existido problemas o 0 en caso de que existiese algún error.

Término	Descripción
lproc	Define a los procesamientos largos de información, estas ejecutarán funciones específicas que realizan cálculos, validaciones e inserciones utilizando una cantidad de datos masivos. En algunos casos el proceso principal llama internamente a otros procesos mediante REST API, los cuales realizan cálculos y actualizaciones de datos que son necesarias para el correcto funcionamiento del proceso principal. La respuesta obtenida puede ser 1, en caso de que no haya existido problemas o 0 en caso de que existiese algún error.
task	Define la ejecución de tareas periódicas que procesan información o realizan respaldos de la misma. La respuesta obtenida puede ser 1, en caso de que no haya existido problemas, o 0 en caso de que existiese algún error.
notif	Genera notificaciones a los usuarios, por lo general son llamados de otros procesos que ya han sido ejecutados. La respuesta obtenida puede ser 1, en caso de que no haya existido problemas o 0 en caso de que existiese algún error.

Términos de objetos

Este término define el tipo de objeto que va ser procesado, consultado o manipulado en la ejecución de la acción antes indicada. Existen términos definidos inicialmente, sin embargo, si se crean nuevos objetos, pueden ser utilizados en este término una vez que haya sido declarado en el URI del sistema REST API. Los términos definidos son los siguientes:

Tabla 7: Términos reservados para objetos

Término	Descripción
Usr	Usuario
Cim	Cita médica
Xla	Examen de laboratorio
Med	Medicamento
Imd	Información médica
Ign	Información general
Par	Parámetros de Sistema
Phm	Historial médico del paciente
msn	Mensaje

Término	Descripción
Rta	Retroalimentación
Not	Notificación
Sec	Seguridad y permisos
Err	Errores del sistema
Aud	Datos para auditoría
Und	No se puede definir

Términos de acción

Determina los métodos que se van a ejecutar en el servidor RES API mediante la petición realizada. Existen términos definidos que representan a las funciones básicas que por lo general se emplean en el procesamiento de datos; siendo estas las siguientes:

Tabla 8: Términos reservados de acciones

Término	Descripción
Ins	Insertar
Con	Consultar

Término	Descripción
Mod	Modificar
Del	Eliminar
Hab	Habilitar
Dhb	Deshabilitar

En caso de que se cree un nuevo método en el sistema, primero hay que registrar el término que va a usar en la URI del sistema REST API, para luego vincularlo al método que se va a ejecutar.

4.2 Análisis de reportes necesarios

Los reportes son una parte muy importante del sistema, debido a que ayudan a agilizar el análisis de datos médicos del paciente para poder realizar un diagnóstico más rápido y eficaz, e inclusive obtener observaciones que puedan ayudar en el inicio de nuevas investigaciones y estudios. Los reportes de mayor utilidad para lograr estos objetivos son los siguientes:

- Reporte de peso promedio agrupado por edad de paciente.
- Reporte de BMI promedio agrupado por edad, estatura y peso del paciente.

- Reporte de retroalimentaciones positivas contra retroalimentaciones negativas obtenidas mensualmente.
- Reporte mensual y semanal de pacientes atendidos por un miembro.
- Reporte semanal de motivos más comunes relacionados a varianzas de carácter negativo en medidas de datos médicos.
- Reporte semanal de tipo de exámenes de laboratorio solicitados.
- Reporte de tiempo promedio de respuesta por tipo de exámenes de laboratorio realizados.
- Reporte de laboratorios comparando su tiempo de respuesta y calidad de resultados.
- Reporte de avances obtenidos mediante el consumo del paciente de los medicamentos diagnosticados.

4.3 Diseño de interfaz gráfica

La interfaz gráfica tiene una arquitectura tipo Responsive, esta arquitectura cuenta con técnicas y métodos que permiten que el sistema pueda ser visualizado correctamente desde cualquier dispositivo. Esta interfaz tendrá un formato reutilizado en las diferentes secciones de los módulos, sin embargo, mediante parámetros que pueden ser configurados se puede

aplicar un template diferente en una URL específica, siendo este template un desarrollo adicional en caso de ser aplicado.

La estructura principal de la interfaz y sus secciones internas son las siguientes:

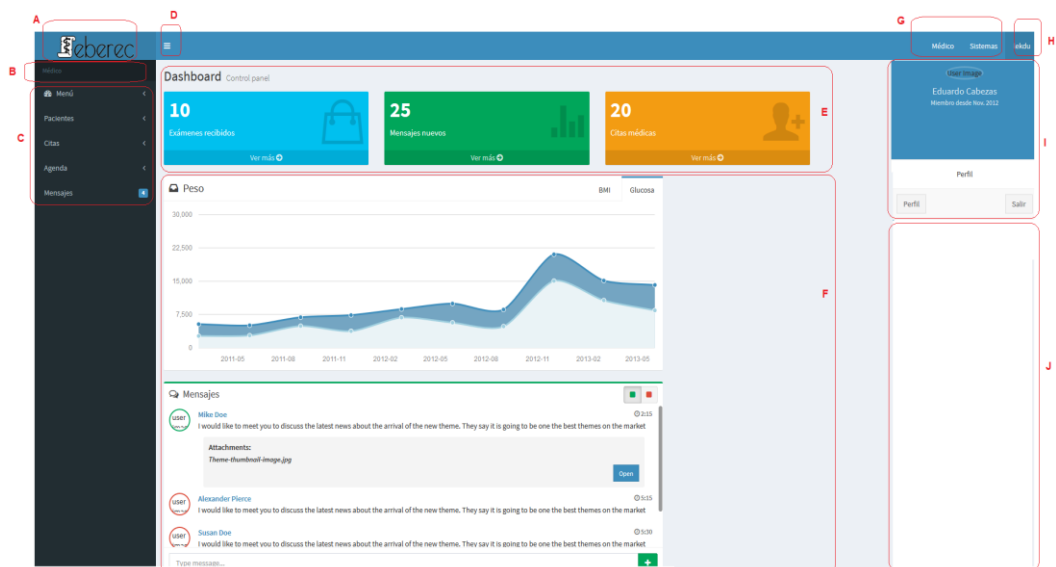


Figura 4.1: Estructura de interfaz gráfica

Fuente: El Autor

Secciones

Las secciones mostradas en la estructura representada en la Figura 4.1, son las siguientes:

- Logo del sistema principal, en este caso el sistema se llama Eberec.
- Nombre del módulo del sistema en el que se encuentra.
- Menú con las opciones disponibles en el módulo y que tenga acceso permitido.

- D. Botón para expandir o comprimir el menú.
- E. Sección de widgets de información, existen widgets predefinidos, pero también se pueden desarrollar nuevos widgets para ser mostrados en esta sección. Esta sección es mostrada solo en la pantalla principal de cada opción de los módulos del sistema.
- F. Contenido, el cual puede ser texto, información, formularios, gráficos, tablas, reportes y agenda. En caso de implementar nuevos templates, estos serán aplicados únicamente en esta sección.
- G. Menú de módulos accesibles.
- H. Link con nombre de usuario, que sirve para mostrar una sección desplegable que muestra información del usuario.
- I. Sección desplegable que muestra información del usuario.
- J. Sección de widgets laterales que muestran información importante del objeto al que se esté consultando, al igual que los widgets de la sección E, también se pueden desarrollar nuevos widgets.

Tipo de contenidos

En la sección F se pueden mostrar uno o varios tipos de contenido al mismo tiempo, dependiendo como esté configurada la opción. Los tipos de

contenido y el tipo de acciones que son usadas para ser visualizadas, son los siguientes:

- **Texto**

Son mensajes, descripciones o información que puede ser mostrada.



Figura 4.2: Vista de tipo de contenido – Texto

Fuente: El Autor

- **Información**

Muestra la información detallada de un objeto específico. Para poder mostrar de una manera adecuada esta información, el sistema analizará de manera automática el resultado obtenido del sistema REST API para interpretarlo y procesarlo para generar la vista final.

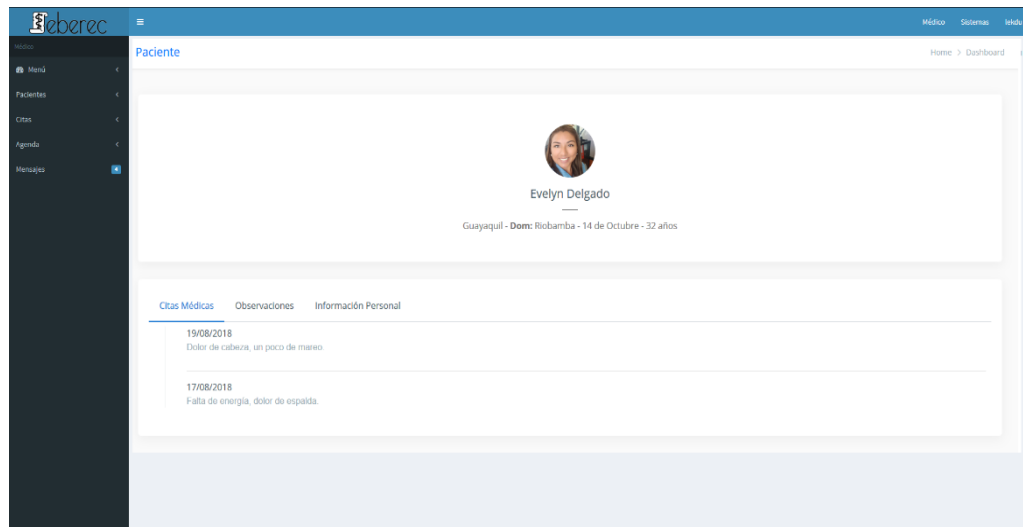


Figura 4.3: Vista de tipo de contenido – Información

Fuente: El Autor

- Formularios

Son utilizados para ingreso o actualización de información de algún objeto del sistema. También es utilizado para el ingreso de retroalimentación o comentarios.

 The screenshot shows the 'Nuevo Paciente' (New Patient) form in the eberec system. The form is titled 'Nuevo Paciente' and is divided into 'Person Info' and 'Domicilio' sections. The 'Person Info' section contains the following fields:

Person Info	
Nombres	Apellidos
<input type="text" value="John doe"/>	<input type="text" value="12n"/>
<small>This is inline help</small>	<small>This field has error.</small>
Género	Fecha de Nacimiento
<input type="text" value="Masculino"/>	<input type="text" value="mm/dd/yyyy"/>
<small>Select your gender</small>	
Domicilio	Lugar de Nacimiento
<input type="text" value="Riobamba"/>	<input type="text" value="Guayaquil"/>
Domicilio	
Dirección	
<input type="text"/>	
Ciudad	Referencia
<input type="text"/>	<input type="text"/>

Figura 4.4: Vista de contenido – Formulario

Fuente: El Autor

- Gráficos

Generalmente son usadas para representar la estadística o los cambios que han sucedido en los valores de la información médica del paciente. También sirven para poder mostrar de una manera fácil de comprender el avance de indicadores con el paso del tiempo. Este tipo de contenido puede ser reutilizado para generar widgets en el sistema. Los gráficos son utilizados de manera obligatoria al momento de presentar reportes como contenido adicional. Estos gráficos pueden ser de tipo barra, pie chart o lineal.



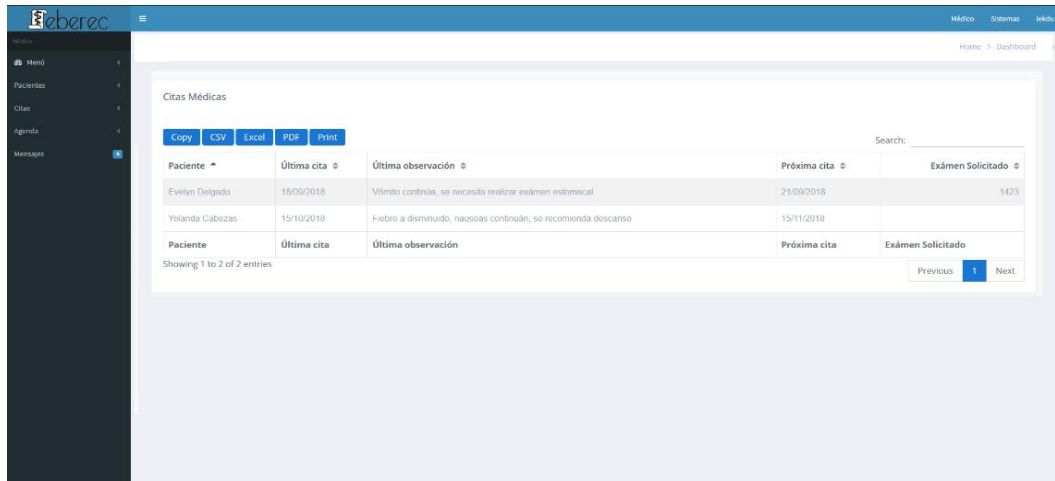
Figura 4.5: Vista de contenido – Gráficos

Fuente: El Autor

- Tablas

Son usadas generalmente para mostrar resultados de consultas cuyo contenido es una lista de varios registros del mismo objeto. Si en los resultados se encuentra información de otros objetos, automáticamente se

generará un hipervínculo que pueda redireccionar a la información de dicho objeto.



The screenshot shows a web application interface with a sidebar menu on the left and a main content area. The main content area displays a table titled "Citas Médicas". Above the table are buttons for "Copy", "CSV", "Excel", "PDF", and "Print". There is also a search bar. The table has five columns: "Paciente", "Última cita", "Última observación", "Próxima cita", and "Examen Solicitado". Two rows of data are visible. Below the table, it says "Showing 1 to 2 of 2 entries" and there are "Previous", "1", and "Next" navigation buttons.

Paciente	Última cita	Última observación	Próxima cita	Examen Solicitado
Evelyn Delgado	18/09/2018	Vómito continúa, se necesita realizar examen estomacal	21/09/2018	1423
Yolanda Cabezas	15/10/2018	Fiebre a disminuido, náuseas continúan, se recomienda descanso	15/11/2018	

Figura 4.6: Vista de contenido – Tablas

Fuente: El Autor

- Reportes

Muestra el cambio de valores en la información médica en una tabla, y adicional se agrega una representación gráfica de dicha información. La información que se va presentar depende de la configuración de los parámetros del reporte. Al igual que con la vista de información, para poder mostrar de una manera adecuada la información, el sistema analizará de manera automática el resultado obtenido del sistema REST API para interpretarlo y procesarlo para generar la vista final.

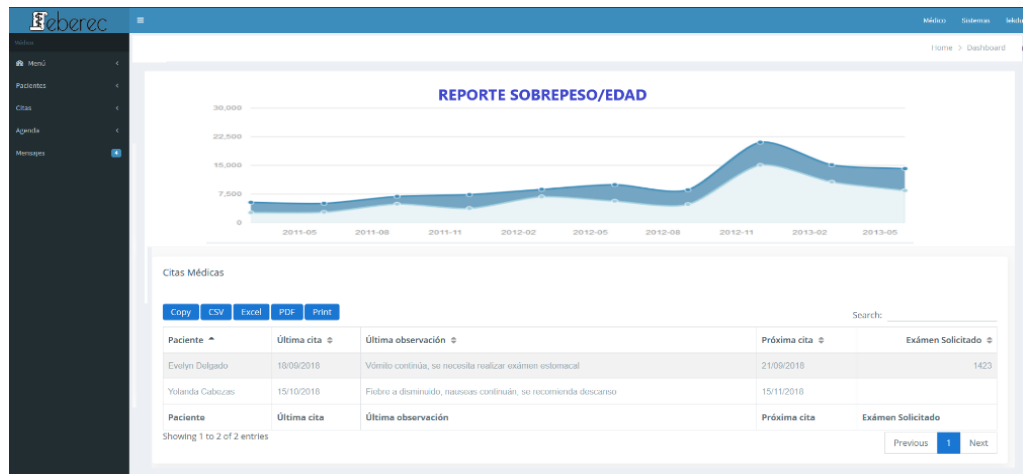


Figura 4.7: Vista de contenido – Reporte

Fuente: El Autor

- Agenda

Esta opción permite administrar de manera visual amistosa, las citas que han sido agendadas, generar nuevas citas o actualizarlas. También se podrán revisar fechas de reuniones o eventos realizados por la sociedad, cuya generación dependerá del administrador del sistema y se agendará de manera automática a todos los miembros de la sociedad.

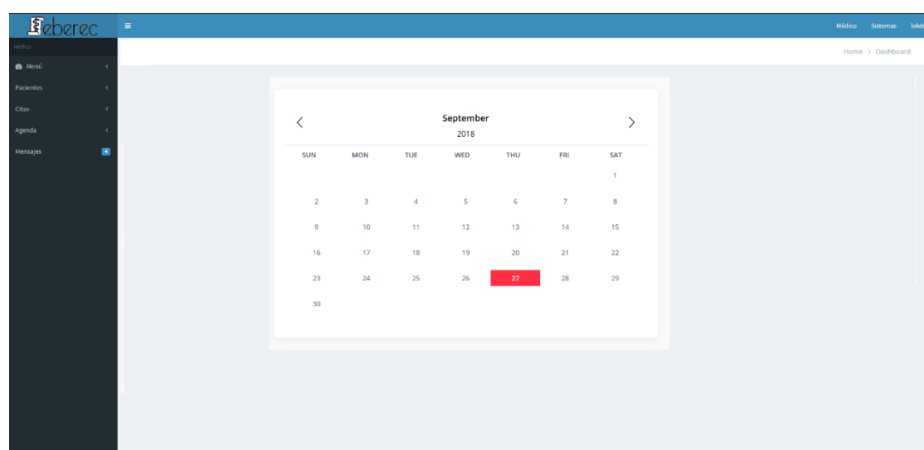


Figura 4.8: Vista de contenido – Agenda

Fuente: El Autor

Pantalla de ingreso al sistema

La única pantalla que es diferente a la estructura principal de la aplicación es la de ingreso al sistema. La cual consiste en un formulario pequeño con el logo del sistema y un fondo diferente.



Figura 4.9: Pantalla de ingreso al sistema

Fuente: El Autor

4.4 Diseño de base de datos

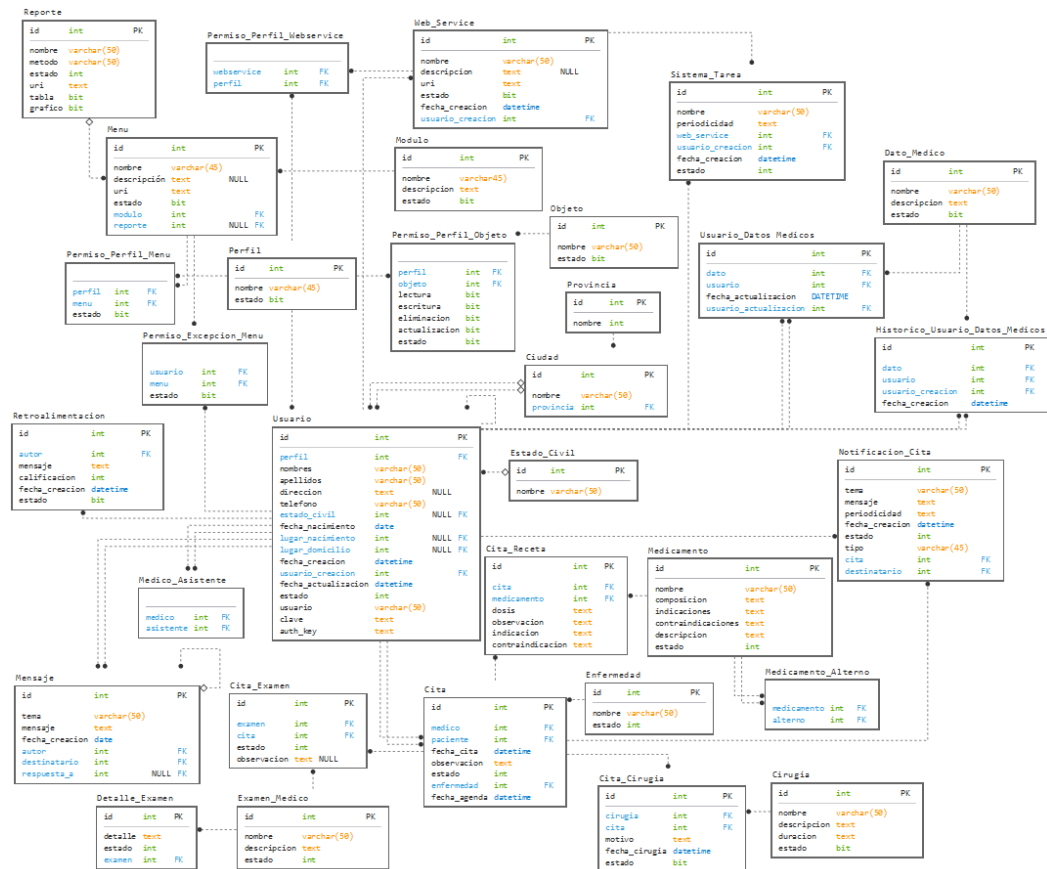


Figura 4.10: Esquema de Base de Datos

Fuente: El Autor

4.5 Diseño de reportes a implementar

Los reportes en el sistema tendrán dos maneras de presentar la información, los cuales también pueden usar ambas presentaciones al mismo tiempo, de ser así configurado. Estas presentaciones son tablas y gráficos estadísticos.

La forma como se van a presentar los reportes depende de la configuración que van a tener dentro del sistema. Primero para poder acceder a un reporte es necesario realizar los pasos para poder crearlo.

Creación de un reporte

Para crear correctamente un reporte se deben seguir los siguientes pasos:



Figura 4.11: Creación de un reporte en el sistema

Fuente: El Autor

1. Diseñar y desarrollar la función en la base de datos que procesará la información y generará los resultados que serán enviados al servidor REST API.
2. Diseñar y desarrollar el método en el sistema REST API que llamará a la función de la base de datos, primero validando si el usuario tiene permisos de ejecución de ese método. En este método se obtendrán los resultados de esta función y los convertirá en objetos JSON, para retornarlos a la aplicación.

3. El administrador del sistema ingresará desde el administrador de parámetros al web service en el sistema, para poder configurar los permisos de ejecución a los perfiles de usuario existentes. Al momento de crear el web service en el sistema, se indicará el tipo de presentaciones que se mostrarán en el reporte.

Estructura de ejecución para generar un reporte

Una vez que se haya creado y configurado el reporte en el sistema, podrá ser generado por los usuarios con perfiles autorizados. Para generar un reporte se ejecutará una estructura de procesos. Esta estructura es representada en la siguiente gráfica:

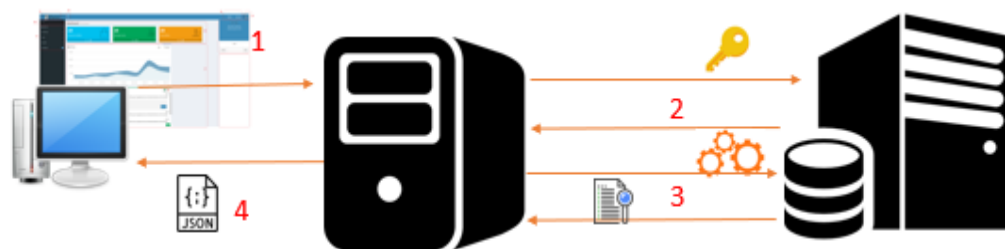


Figura 4.12: Estructura de ejecución para generar un reporte

Fuente: El Autor

1. El sistema web realizará una petición a una URI designada a un reporte en el sistema REST API, adjuntando un web token de acceso en la cabecera.
2. El sistema REST API procesará la petición, validará el web token y se realizará una petición a la base de datos para verificar si el perfil del usuario tiene permiso para ejecutar la función del proceso.

3. Si el usuario tiene permiso para acceder al reporte, se ejecutará la función de la base de datos para obtener la información necesaria para generar el reporte. Al mismo tiempo se obtendrá la configuración de presentación del reporte.
4. La información y la configuración del reporte son convertidos en datos de formato JSON en el servidor REST API, para luego ser enviados al servidor web que interpretará dicho resultado para presentarlo en la interfaz del sistema al usuario; en caso de tener ambos tipos de presentación, se presentará el gráfico estadístico y la tabla de datos, en ese orden.

Todo reporte permitirá la exportación de su información en archivos de formato PDF, CSV y Excel; al momento de realizar una exportación se generará un registro en los archivos de logs.

CAPÍTULO 5

DESARROLLO Y PRUEBAS

5.1 Desarrollo de la aplicación REST

La aplicación REST está desarrollada en lenguaje de programación Python, versión 3.6, utilizando el framework Django REST, versión 3. Este framework utiliza la arquitectura web MVT. Este sistema tendrá los modelos de cada objeto existente en el proyecto, con los mismos atributos a los campos de la base de datos, el motivo de esto es porque Django REST se sincroniza y crea la estructura de la base de datos dependiendo de los modelos que hayan sido creados en el sistema y de los atributos

que se les asigne; esto mediante comandos ejecutados en el panel de Python. Estos módulos pueden ser categorizados mediante aplicaciones que se crean dentro del sistema, ya sea de manera manual o mediante línea de comandos.

En este sistema existen dos tipos de aplicaciones, seguridad y entidades. La aplicación de seguridad tendrá los modelos y vistas encargados de realizar procesos.

Las funciones que son desarrolladas en los modelos del sistema sirven únicamente para los procesos que se van a ejecutar en la base de datos. En caso de inserciones, actualizaciones y eliminaciones, se utilizan las funciones ORM nativas de Django REST para la ejecución de estos procesos de la información. Si se van a realizar consultas o procesos complejos en la base de datos, entonces se utiliza la función nativa de Django REST para ejecutar sentencias SQL o para llamar a funciones almacenadas en la base de datos.

En la capa “views” del sistema se desarrollarán los métodos del web service que se está implementando, los cuales llaman a funciones de módulos para poder interactuar con la base de datos. Estos métodos serán ejecutados mediante pedidos HTTP que recibirá el sistema en sus URIs. Estos están definidos en la sección “url” del sistema REST, en donde también se indica el método que va a ejecutar cada URI. Cada método

recibirá parámetros tipo POST desde la petición del servidor de origen, esto sirve para la correcta ejecución de las funciones almacenadas de la base de datos o para las acciones a realizar con los modelos del sistema en la base de datos. Los resultados que cada método va a retornar como respuesta al servidor de origen de la petición, tienen formato JSON.

Antes de ejecutar cada método el sistema verificará la validez del web token que es enviado desde el servidor de origen, esto se realiza como medida de seguridad del sistema.

Lo único que retorna este sistema son respuestas de información en formato JSON, por lo que la capa “template” de este sistema no es manipulada y su única función es mostrar estos resultados para que sean capturados por el sistema web o por cualquier sistema que lo esté consumiendo.

5.2 Desarrollo de la aplicación web

La aplicación web está desarrollada en lenguaje de programación Python, versión 3.6, utilizando el framework Django, versión 2.1.1. Este framework utiliza la arquitectura web MVT. Al igual que en el sistema REST, este sistema tendrá los modelos de cada objeto existente en el proyecto, con los mismos atributos a los campos de la base de datos, con la diferencia que los modelos de este sistema tienen funciones que realizan peticiones únicamente al sistema REST y no a la base de datos.

Este sistema define las funciones que capturarán a los parámetros que son enviados desde la interfaz en la capa “views”, estas analizarán y validarán estos parámetros para que sean procesados mediante las funciones que se encuentran definidas en los diferentes modelos creados en el sistema. Al final de la función de la capa “views” se ejecutará una función que envíe los parámetros al template, la cual presenta la información en el sistema al usuario.

En este sistema existirán 2 tipos de validaciones para datos de formularios; la primera validación se la realizará a nivel de interfaz mediante el uso de funciones desarrolladas con la librería javascript jQuery que sirven para verificar que la información obligatoria haya sido ingresada, limitación de ingreso de tipo de datos incorrectos en los campos, comprobar que los datos tengan formato correcto, entre otros; este tipo de validaciones pueden ser indicadas al momento de declarar el atributo de un modelo en el sistema como un parámetro adicional. La segunda validación se la realizará en la capa de “views” dentro de la función que se vaya a ejecutar, esta realizará el mismo tipo de validaciones que se realizan en la interfaz.

5.3 Pruebas internas

Las pruebas internas se las realizaron mediante la ejecución de los sistemas en un ambiente virtual local de Python utilizando la librería vEnv. Este ambiente virtual es portable y mantendrá toda librería de Python que

haya sido instalada en el mismo. Una vez listo el ambiente se ejecutan ambos proyectos mediante consola, indicando que el sistema web se ejecute en el puerto local 8000 y el sistema REST en el puerto 8001.

Una vez que los proyectos hayan sido ejecutados, se ingresará al sistema mediante la interfaz del ambiente web y se hacen las pruebas necesarias.

Las pruebas internas que se realizaron fueron las siguientes:

Tabla 9: Pruebas internas

Prueba	Resultado
Usuario Administrador	
Creación y edición de objetos, entre los cuales están:	La creación y edición de los diferentes objetos fueron realizados de manera correcta.
- Usuario perfil médico	Médico: Eduardo Cabezas.
- Laboratorio	Laboratorio: Sana Sana.
- Usuario perfil laboratorista	Laboratorista: Verónica Cantuna. Administrador: Luis Cabezas.
- Usuario perfil	

Prueba	Resultado
administrador - Usuario perfil auditor	Auditor: Evelyn Delgado.
Edición de permisos	Se realiza cambios de permisos de usuario, quitando la opción de envío de mensajes al perfil médico; se verificó que un usuario de ese perfil ya no contaba con ese permiso y se procedió a revertir el cambio de permisos.
Descarga de logs	Los logs se pudieron descargar exitosamente obteniendo un archivo comprimido en formato zip con el archivo de logs.
Deshabilitar usuario	El usuario se deshabilitó correctamente, se realizó la prueba de ingreso al sistema con dicho usuario y este no tuvo éxito.
Rehabilitar usuario	Se habilitó correctamente al usuario, se pudo acceder al sistema con dicho usuario.

Prueba	Resultado
Usuario Médico	
Creación y edición de paciente	Se creó correctamente al paciente Ernesto Roldán y se editaron sus datos personales con éxito.
Agenda y creación de cita	Se agendó correctamente la cita.
Creación y edición de medicamento	Se creó correctamente el medicamento Arcoxia, y se editaron sus componentes.
Atención de cita, creación de receta, examen médico, cirugía, creación de notificaciones y alarmas	Se atendió a la cita del paciente, se recetó tomar Arcoxia una vez al día, se agendó una cirugía en la pierna. Se crearon correctamente las notificaciones de la cita realizada y los recordatorios del consumo del medicamento.
Agendar cita con otro médico miembro	Se agendó una cita médica con el médico Marco Martínez.

Prueba	Resultado
Creación de asistente	Se creó al asistente José Rodríguez.
Deshabilitar asistente	Se deshabilitó al asistente de manera correcta, este ya no pudo acceder al sistema.
Rehabilitar asistente	Se rehabilitó al asistente de manera correcta, pudo acceder al sistema exitosamente.
Revisar resultados de exámenes de laboratorio	El médico pudo revisar los resultados de su examen de laboratorio. Previamente el usuario laboratorista respondió la solicitud de examen.
Ingresar retroalimentación	Se ingresó correctamente la retroalimentación al sistema.
Enviar mensaje a otro usuario	Se envió un mensaje al miembro Marco Martínez.

Prueba	Resultado
Usuario Paciente	
Consulta de citas realizadas	El paciente Ernesto Roldán pudo consultar su cita con el Dr. Eduardo Cabezas.
Visualizar información de cita médica	Se pudo conocer la información de la cita médica. Se recetó tomar Arcoxia una vez al día, se agendó una cirugía en la pierna.
Consulta de datos médicos	Se pudo visualizar correctamente la información médica del paciente.
Consulta de recomendaciones de médicos	Se visualizaron las recomendaciones realizadas por el médico de forma correcta.
Calificar médico	Se calificó correctamente al médico con 5 estrellas.
Visualización de recordatorios	Se visualizó el recordatorio de tomar Arcoxia una vez al día.

Prueba	Resultado
Revisar resultados de exámenes de laboratorio	El paciente pudo revisar los resultados de su examen de laboratorio. Previamente el usuario laboratorista respondió la solicitud de examen.
Ingreso de retroalimentación	Se ingresó correctamente la retroalimentación al sistema.
Enviar mensaje a otro usuario	Se envió un mensaje al médico Eduardo Cabezas.
Usuario Laboratorista	
Revisar solicitudes de exámenes de laboratorio	Se visualizaron las solicitudes de los exámenes médicos correctamente.
Responder solicitud de examen de laboratorio	Se envió la respuesta del examen de Ernesto Roldán. Previamente el paciente tuvo que acercarse al laboratorio para poder tomar las muestras para realizar los exámenes de laboratorio.

Prueba	Resultado
Enviar mensaje a otro usuario	Se envió un mensaje al médico Eduardo Cabezas correctamente.
Usuario Auditor	
Descarga de logs	Los logs se pudieron descargar exitosamente obteniendo un archivo comprimido en formato zip con el archivo de logs.
Ingreso de retroalimentación	Se ingresó correctamente la retroalimentación al sistema.
Enviar mensaje a otro usuario	Se envió un mensaje al médico Eduardo Cabezas de forma correcta.

5.4 Pruebas de stress

Las pruebas de estrés se las realizaron mediante una aplicación web desarrollada en Python y que utiliza el framework LocustIO. Para la ejecución de la aplicación es necesario que se cree únicamente un archivo

que contenga las clases que definan las tareas, las cuales indican la URI que se va a llamar al momento de ejecutar la aplicación y la cantidad de hilos que se desee que se generen al momento en que la aplicación realice la prueba de estrés.

```
from locust import HttpLocust, TaskSet, task

class UserBehavior(TaskSet):
    def on_start(self):
        """ on_start is called when a Locust start before any task is scheduled """
        self.login()

    def on_stop(self):
        """ on_stop is called when the TaskSet is stopping """
        self.logout()

    def login(self):
        self.client.post("/login", {"username":"ellen_key", "password":"education"})

    def logout(self):
        self.client.post("/logout", {"username":"ellen_key", "password":"education"})

    @task(2)
    def index(self):
        self.client.get("/")

    @task(1)
    def profile(self):
        self.client.get("/profile")

class WebsiteUser(HttpLocust):
    task_set = UserBehavior
    min_wait = 5000
    max_wait = 9000
```

Figura 5.1: Código de ejemplo de LocustIO

Fuente: El Autor

Para iniciar la aplicación es necesario ejecutar el siguiente comando en la consola de Python:

```
locust --host=dominio:Puerto1
```

```
locust -f aplicacion.py --master --host=http://otroDominio:puerto2
```

Donde dominio y puerto1 son el dominio y el puerto donde se podrán acceder a la aplicación desde el navegador web; otroDominio y puerto2 son el dominio y el puerto donde se va a realizar la prueba de estrés.

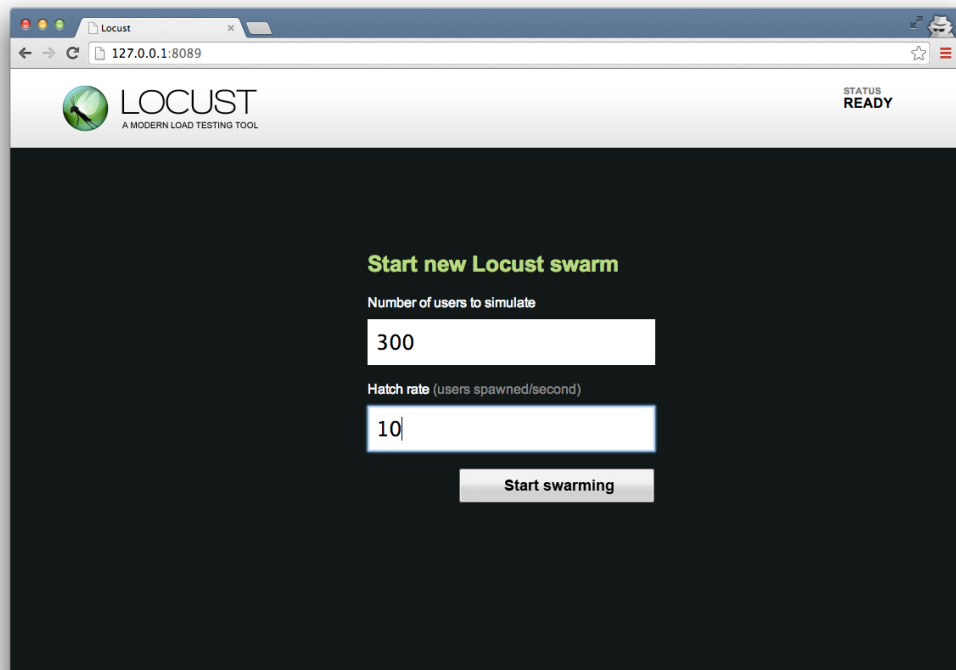


Figura 5.2: Pantalla de la aplicación LocustIO

Fuente: El Autor

Una vez inicializada la aplicación, esta podrá ser accedida desde el navegador web utilizando el dominio y el puerto que fueron indicadas en la aplicación. En esta pantalla se puede indicar la cantidad de usuarios que se va simular en el acceso al sistema destinatario; el cual representarían las variables otroDominio y puerto2 que fueron antes mencionadas; y el rango de tiempo en que parcialmente van a ser generados, es decir, si se

ingresa 100 en el primer campo y 10 en el segundo campo, se van a generar 10 usuarios a cada minuto hasta generar 100 usuarios a simular.

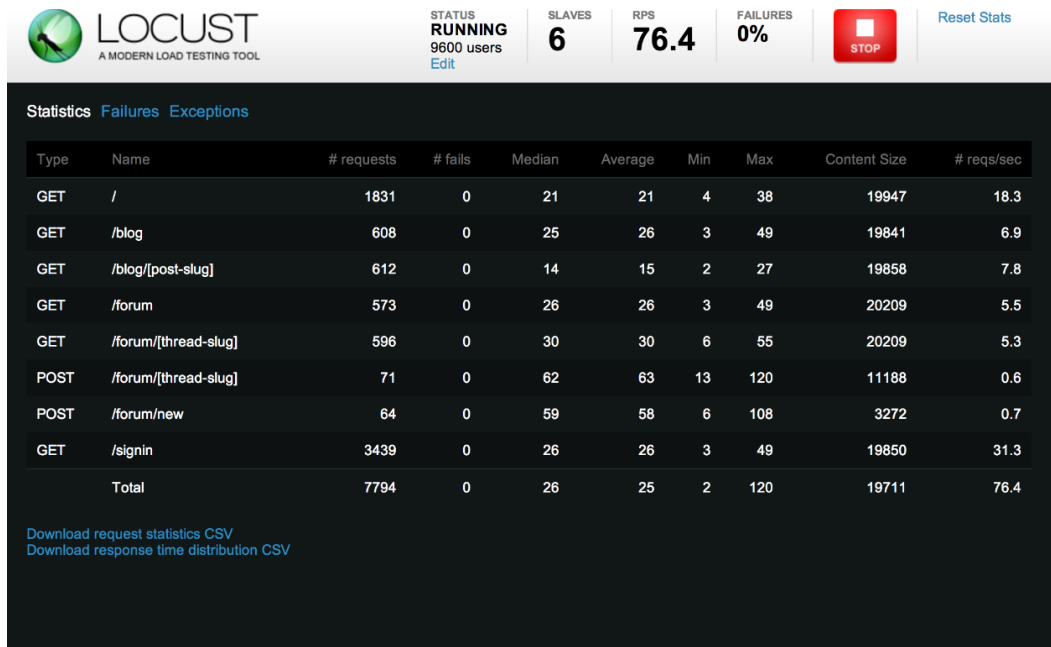


Figura 5.3: Pantalla de resultados de LocustIO

Fuente: El Autor

Después de que la aplicación realice la prueba de estrés, se visualizarán los resultados en la misma. En estos resultados estarán:

- El tipo de petición que se haya realizado.
- La URI.
- El número de segmentos procesados.
- El número de peticiones que hayan sido fallidas.
- La mediana del tiempo de procesamiento.

- El promedio del tiempo de procesamiento.
- El tiempo mínimo de procesamiento.
- El tiempo máximo de procesamiento.
- El tamaño del contenido.
- El número de peticiones por segundo.

5.5 Pruebas con usuarios

Para las pruebas con los usuarios se publicaron los dos sistemas y la base de datos, de manera interna en tres servidores locales. Solo miembros de la Sociedad de Trauma tuvieron acceso a los sistemas. Para las pruebas con pacientes, el médico les pedía que lo utilizaran en su equipo una vez terminada la cita médica, recibiendo al mismo tiempo retroalimentación directa por parte de los pacientes y recomendaciones para realizar mejoras.

El tipo de pruebas que se realizaron con los usuarios fueron las mismas que se hicieron de manera interna, obteniendo los mismos resultados.

Cabe mencionar que antes de realizar las pruebas del sistema con el usuario, se realizaron mediciones de los tiempos de trabajo en las tareas diarias que realizaban y que tenían relación con el sistema que se está desarrollando. Esto se realiza con la finalidad de tener una medida inicial

del tiempo de trabajo de cada tarea. Estas pruebas y mediciones se las ejecutaron con 7 de los 25 miembros de la Sociedad de Trauma – Capítulo Chimborazo.

Tabla 10: Pruebas con usuarios

Prueba	Resultado
Usuario Administrador – Luis Eduardo Cabezas	
Creación y edición de objetos, entre los cuales están: <ul style="list-style-type: none"> - Usuario perfil médico - Laboratorio - Usuario perfil laboratorista - Usuario perfil administrador - Usuario perfil auditor 	La creación y edición de los diferentes objetos fueron realizados de manera correcta. Médico: Eduardo Cabezas. Laboratorio: Sana Sana. Laboratorista: Verónica Cantuna. Administrador: Luis Cabezas. Auditor: Evelyn Delgado.

Prueba	Resultado
Edición de permisos	Se realiza cambios de permisos de usuario, quitando la opción de envío de mensajes al perfil médico; se verificó que un usuario de ese perfil ya no contaba con ese permiso y se procedió a reversar el cambio de permisos.
Descarga de logs	Los logs se pudieron descargar exitosamente obteniendo un archivo comprimido en formato zip con el archivo de logs.
Deshabilitar usuario	El usuario se deshabilitó correctamente, se realizó la prueba de ingreso al sistema con dicho usuario y este no tuvo éxito.
Rehabilitar usuario	Se habilitó correctamente al usuario, se pudo acceder al sistema con dicho usuario.
Usuario Médico – Dr. Eduardo Cabezas Cevallos	

Prueba	Resultado
Creación y edición de paciente	Se creó correctamente al paciente Ernesto Roldán y se editaron sus datos personales con éxito.
Agenda y creación de cita	Se agendó correctamente la cita.
Creación y edición de medicamento	Se creó correctamente el medicamento Arcoxia, y se editaron sus componentes.
Atención de cita, creación de receta, examen médico, cirugía, creación de notificaciones y alarmas	Se atendió a la cita del paciente, se recetó tomar Arcoxia una vez al día, se agendó una cirugía en la pierna. Se crearon correctamente las notificaciones de la cita realizada y los recordatorios del consumo del medicamento.
Agendar cita con otro médico miembro	Se agendó una cita médica con el médico Marco Martínez.
Creación de asistente	Se creó al asistente José Rodríguez.

Prueba	Resultado
Deshabilitar asistente	Se deshabilitó al asistente de manera correcta, este ya no pudo acceder al sistema.
Rehabilitar asistente	Se rehabilitó al asistente de manera correcta, pudo acceder al sistema exitosamente.
Revisar resultados de exámenes de laboratorio	El médico pudo revisar los resultados de su examen de laboratorio. Previamente el usuario laboratorista respondió la solicitud de examen.
Ingresar retroalimentación	Se ingresó correctamente la retroalimentación al sistema.
Enviar mensaje a otro usuario	Se envió un mensaje al miembro Marco Martínez.
Usuario Paciente – Ernesto Roldán	
Consulta de citas	El paciente Ernesto Roldán pudo consultar

Prueba	Resultado
realizadas	su cita con el Dr. Eduardo Cabezas.
Visualizar información de cita médica	Se pudo conocer la información de la cita médica. Se recetó tomar Arcoxia una vez al día, se agendó una cirugía en la pierna.
Consulta de datos medicos	Se pudo visualizar correctamente la información médica del paciente.
Consulta de recomendaciones de médicos	Se visualizaron las recomendaciones realizadas por el médico correctamente.
Calificar medico	Se calificó correctamente al médico con 5 estrellas.
Visualización de recordatorios	Se visualizó el recordatorio de tomar Arcoxia una vez al día.
Revisar resultados de exámenes de laboratorio	El paciente pudo revisar los resultados de su examen de laboratorio. Previamente el usuario laboratorista respondió la solicitud

Prueba	Resultado
	de examen.
Ingreso de retroalimentación	Se ingresó correctamente la retroalimentación al sistema.
Enviar mensaje a otro usuario	Se envió un mensaje al médico Eduardo Cabezas.
Usuario Laboratorista – Dra. Verónica Cantuna	
Revisar solicitudes de exámenes de laboratorio	Se visualizaron las solicitudes de los exámenes médicos correctamente.
Responder solicitud de examen de laboratorio	Se envió la respuesta del examen de Ernesto Roldán. Previamente el paciente tuvo que acercarse al laboratorio para poder tomar las muestras para realizar los exámenes de laboratorio.
Enviar mensaje a otro usuario	Se envió un mensaje al médico Eduardo Cabezas correctamente.

Prueba	Resultado
Usuario Auditor – Ing. Evelyn Delgado Soto	
Descarga de logs	Los logs se pudieron descargar exitosamente obteniendo un archivo comprimido en formato zip con el archivo de logs.
Ingreso de retroalimentación	Se ingresó correctamente la retroalimentación al sistema.
Enviar mensaje a otro usuario	Se envió un mensaje al médico Eduardo Cabezas correctamente.

5.6 Implementación del sistema

La implementación del sistema se realizará mediante un hosting tercerizado, el cual debe permitir la publicación de aplicaciones web desarrolladas en Python. En este hosting se administrará dos dominios, uno para el sistema web y otro para el sistema REST y para la base de datos, brindando la seguridad adecuada para la base de datos. Estos dominios van a ser públicos. Al momento de ingresar al sistema web se

mostrará directamente la pantalla de ingreso al sistema. Si se ingresa al dominio del sistema REST como pantalla principal, se mostrará una pantalla estática que no tenga relación con el sistema REST o web, esto como medida de seguridad.

CAPÍTULO 6

ANÁLISIS DE RESULTADOS

6.1 Análisis de tiempos utilizados en citas médicas antes y después de usar el sistema

Como se mencionó anteriormente, se realizaron mediciones del tiempo de atención a los pacientes antes y después de usar el sistema. Para poder realizar estas mediciones se acordó con 7 miembros de la Sociedad de Trauma para que inicialmente se midieran los tiempos de atención a los 5 primeros pacientes que ingresaran en su consultorio en un determinado día, se realizaría este procedimiento a cada miembro en distintos días;

estos pacientes tendrían que ser citados una semana después para otra revisión. En la segunda revisión se realizaría la atención al paciente utilizando el sistema implementado. Después de haberse realizado estas mediciones se pudieron obtener los siguientes resultados:

MEDICO	Tarea	1		2		3		4		5		Antes	Después
		Antes	Después	Antes	Después	Antes	Después	Antes	Después	Antes	Después		
Geovany Cazorla	Registro/consulta de datos de usuario	5	3	6	3	7	4	5	4	5	5	105	62
	Diagnóstico	7	5	10	7	16	10	17	11	15	8		
	Solicitud de exámenes médicos	0	0	5	1	0	0	0	0	7	1		
	Total	12	8	21	11	23	14	22	15	27	14		
Victor Quiziguña	Registro/consulta de datos de usuario	6	4	5	3	6	5	7	3	6	3	119	69
	Diagnóstico	15	10	8	5	17	11	15	9	20	13		
	Solicitud de exámenes médicos	5	1	4	1	0	0	5	1	0	0		
	Total	26	15	17	9	23	16	27	13	26	16		
Marco Martínez	Registro/consulta de datos de usuario	4	4	5	3	6	4	5	5	3	3	81	50
	Diagnóstico	16	11	10	8	13	9						
	Solicitud de exámenes médicos	6	1	0	0	5	1	0	0	8	1		
	Total	26	16	15	11	24	14	5	5	11	4		
Fabian Cruz	Registro/consulta de datos de usuario	5	4	6	3	5	3	7	4	5	2	102	58
	Diagnóstico	20	12	18	12	9	9	7	3	8	4		
	Solicitud de exámenes médicos	0	0	7	1	0	0	0	0	5	1		
	Total	25	16	31	16	14	12	14	7	18	7		
Arturo Cantuña	Registro/consulta de datos de usuario	6	3	5	3	8	5	4	4	5	3	122	69
	Diagnóstico	10	7	15	10	17	10	20	14	10	7		
	Solicitud de exámenes médicos	10	1	0	0	5	1	7	1	0	0		
	Total	26	11	20	13	30	16	31	19	15	10		
Aurora Alvarado	Registro/consulta de datos de usuario	7	3	7	4	5	3	8	5	5	3	105	62
	Diagnóstico	14	10	14	10	18	10	11	7	10	6		
	Solicitud de exámenes médicos	0	0	0	0	0	0	0	0	6	1		
	Total	21	13	21	14	23	13	19	12	21	10		
Flavio Vintimilla	Registro/consulta de datos de usuario	5	4	7	3	5	3	6	4	7	5	122	77
	Diagnóstico	12	8	15	11	20	14	17	12	16	11		
	Solicitud de exámenes médicos	5	1	0	0	0	0	7	1	0	0		
	Total	22	13	22	14	25	17	30	17	23	16		
Irian Naranjo	Registro/consulta de datos de usuario	7	3	6	3	5	4	7	5	5	5	74	51
	Diagnóstico	8	5	10	7	15	12	14	10	20	14		
	Solicitud de exámenes médicos	0	0										
	Total	15	8	10	7	15	12	14	10	20	14		
											830	498	
											Ahorro		332
											Ahorro Promedio por Médico		47,42857143

Figura 6.1: Tiempos de atención (minutos) antes y después de aplicar el sistema

Fuente: El Autor

En base al análisis de los datos registrados se pudo obtener que para los 7 miembros atendiendo 5 pacientes cada uno, se han disminuido alrededor de 332 minutos de atención en total, o alrededor de 47 minutos por médico.

6.2 Mejora en desempeño por parte de médicos

Tomando en cuenta la información presentada en la Figura 6.1, se puede ver en este proceso que los 7 miembros de la Sociedad de Trauma que actuaron en la medición de tiempos de atención al paciente, han empleado 830 minutos de atención antes de usar el sistema y 498 minutos de atención después de utilizar el sistema, por lo que se puede concluir que el desempeño de estos médicos ha incrementado en un 40% en la atención médica que se ha brindado a los pacientes mediante el uso del sistema.

6.3 Mejora en satisfacción del servicio brindado a los pacientes

En el proceso de medición de tiempos de atención a los 7 miembros de la Sociedad de Trauma, se accedió también a realizar una encuesta a los 5 pacientes de cada médico utilizando como herramientas a Google Forms y una tablet.

Para la elección de los temas a tratar en el formulario se tomaron en cuenta los puntos de opinión del paciente de mayor interés por parte de los miembros de la Sociedad de Trauma, y que fue el cimiento para los

objetivos y funciones del sistema desarrollado. El formulario es el siguiente:

El tiempo de consulta médica es adecuado? *

	1	2	3	4	5	
Mala	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Buena

Tiempo en obtener historial médico es adecuado? *

	1	2	3	4	5	
Mala	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Buena

Tiempo de consulta con exámen de laboratorio es adecuado? *

	1	2	3	4	5	
Mala	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Buena

Recuerda fácilmente los medicamentos diagnosticados y su consumo? *

	1	2	3	4	5	
Mala	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Buena

Facilidad de conseguir forma de contacto con el médico. *

	1	2	3	4	5	
Mala	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Buena

Cuántas veces cada tres meses, por lo general, ha perdido el documento para solicitar exámenes médicos? *

1. 0
2. 1
3. 2
4. 3
5. 4
6. 5
7. Más

Tiempo en consultar fecha de cita agendada, en caso de olvido. *

	1	2	3	4	5	
Mala	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Buena

Figura 6.2: Encuesta realizada a los pacientes para medir nivel de satisfacción

Fuente: El Autor

Los resultados obtenidos de la encuesta antes del uso del sistema son los siguientes:

El tiempo de consulta médica es adecuado?

35 respuestas

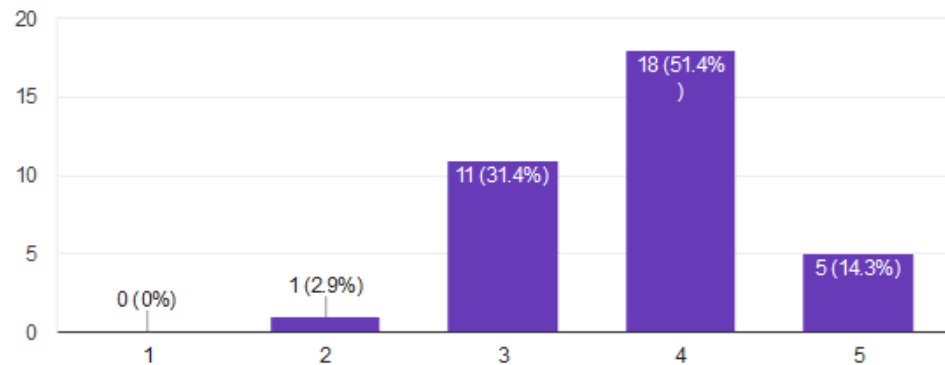


Figura 6.3: Resultado de encuesta antes del sistema - Pregunta 1

Fuente: El Autor

Tiempo en obtener historial médico es adecuado?

35 respuestas

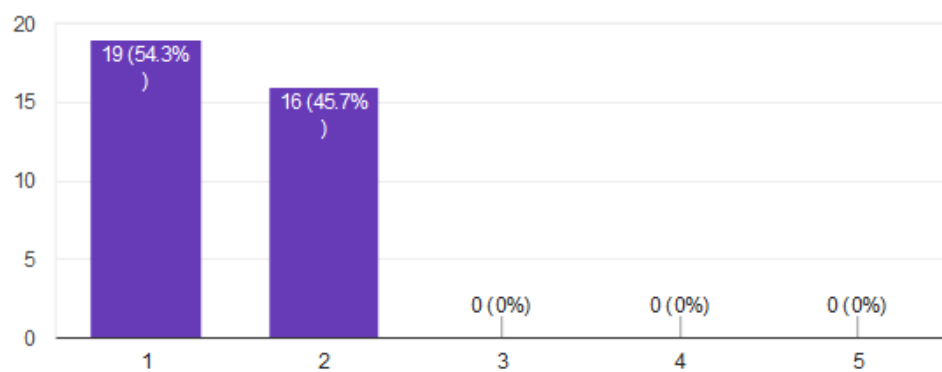


Figura 6.4: Resultado de encuesta antes del sistema - Pregunta 2

Fuente: El Autor

Tiempo de consulta con exámen de laboratorio es adecuado?

35 respuestas

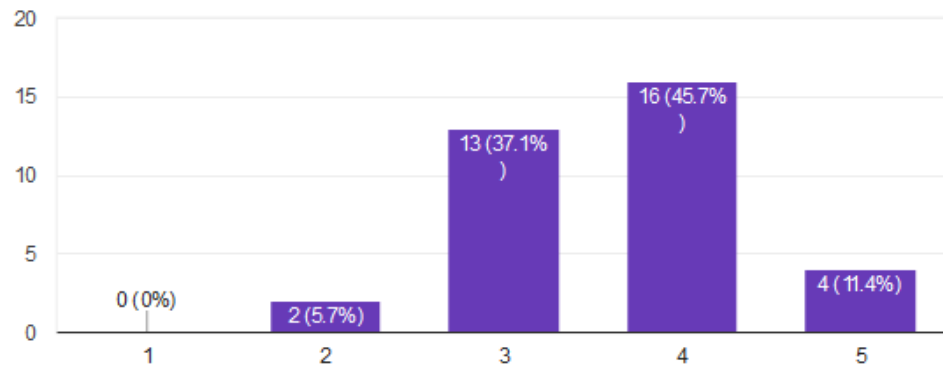


Figura 6.5: Resultado de encuesta antes del sistema - Pregunta 3

Fuente: El Autor

Recuerda fácilmente los medicamentos diagnosticados y su consumo?

35 respuestas

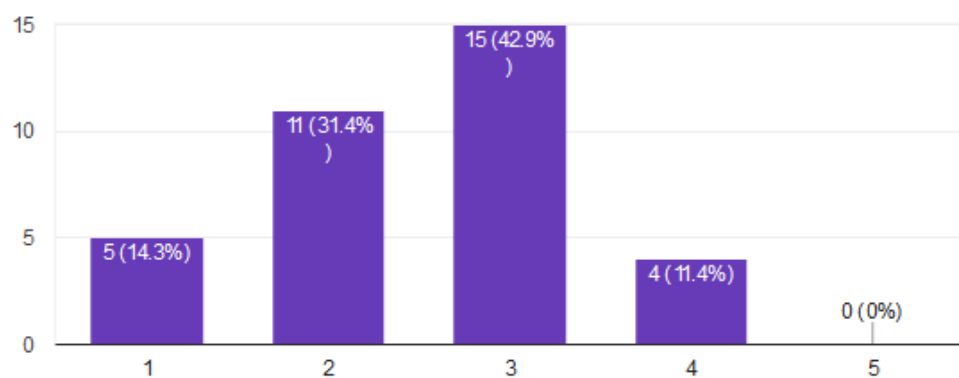


Figura 6.6: Resultado de encuesta antes del sistema - Pregunta 4

Fuente: El Autor

Facilidad de conseguir forma de contacto con el médico.

35 responses

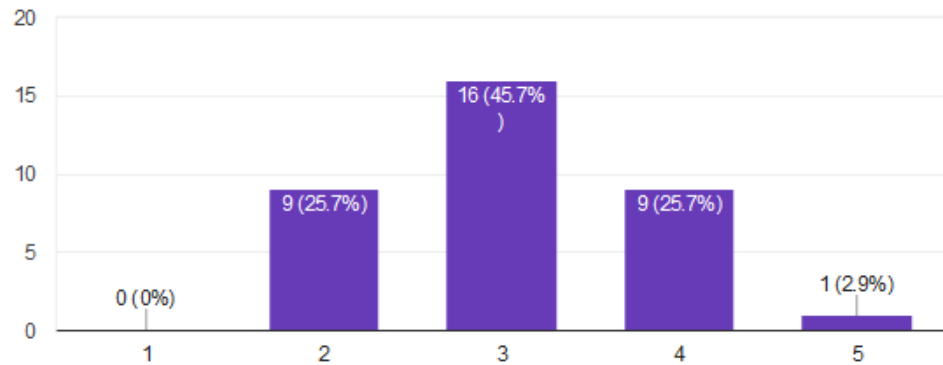


Figura 6.7: Resultado de encuesta antes del sistema - Pregunta 5

Fuente: El Autor

Cuántas veces cada tres meses, por lo general, ha perdido el documento para solicitar exámenes médicos?

35 responses

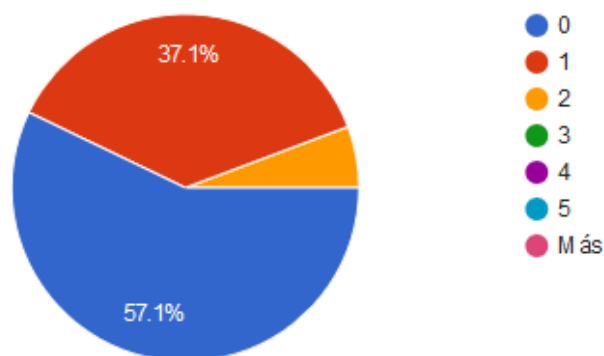


Figura 6.8: Resultado de encuesta antes del sistema - Pregunta 6

Fuente: El Autor

Tiempo en consultar fecha de cita agendada, en caso de olvido.

35 responses

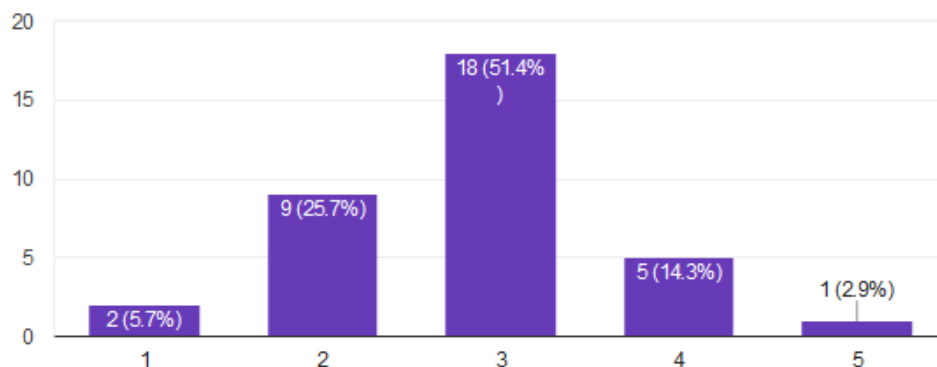


Figura 6.9: Resultado de encuesta antes del sistema - Pregunta 7

Fuente: El Autor

Analizando la información obtenida mediante las respuestas a las preguntas realizadas a los pacientes, se puede observar que:

- El mayor problema que se presenta es la falta de satisfacción con el tiempo para poder obtener el historial médico.
- La manera de solicitar y obtener los resultados de los exámenes de laboratorio. En esta encuesta se puede observar que existen casos en que el paciente ha perdido el documento de solicitud de examen de laboratorio brindado por el médico.
- Los tiempos de atención por parte del médico tiene una aprobación promedio para el paciente.
- Existen dificultades para que el paciente recuerde su medicación y el

consumo de la misma, esto puede suceder en caso de perder la receta médica.

- Existe un poco de dificultad para poder conseguir información para contactar al médico o para consultar futuras consultas.

En la segunda encuesta, al momento de realizar la consulta médica mediante el sistema, se pudieron obtener los siguientes resultados:

El tiempo de consulta médica es adecuado?

35 respuestas

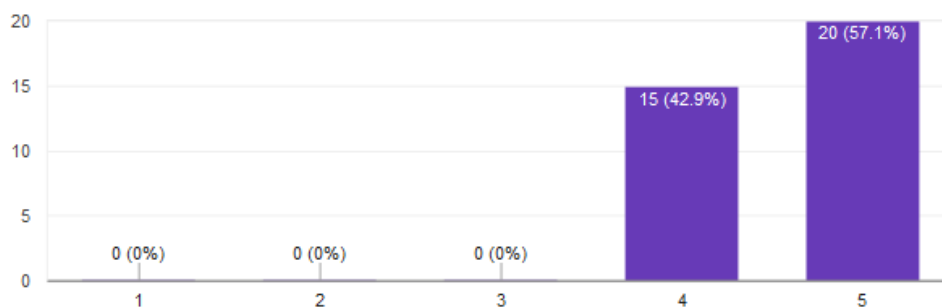


Figura 6.10: Resultado de encuesta después del sistema–Pregunta 1

Fuente: El Autor

Tiempo en obtener historial médico es adecuado?

35 respuestas

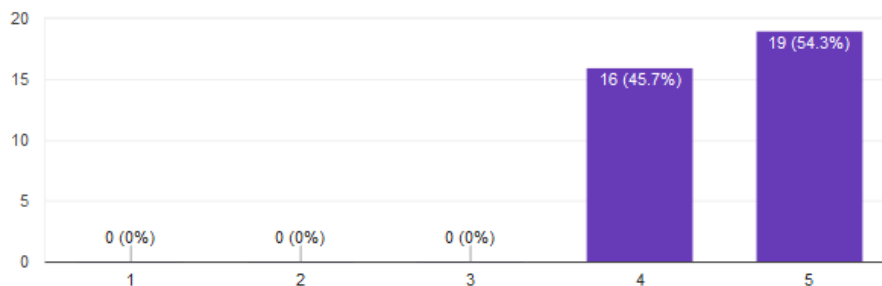


Figura 6.11: Resultado de encuesta después del sistema–Pregunta 2

Fuente: El Autor

Tiempo de consulta con exámen de laboratorio es adecuado?

35 responses

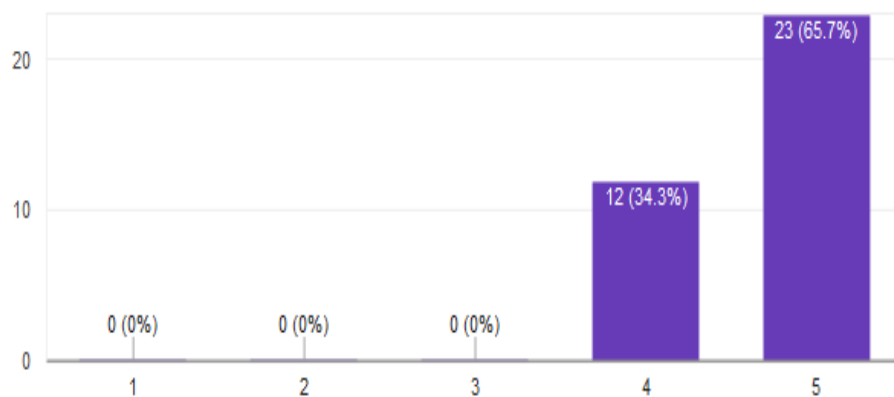


Figura 6.12: Resultado de encuesta después del sistema–Pregunta 3

Fuente: El Autor

Recuerda fácilmente los medicamentos diagnosticados y su consumo?

35 responses

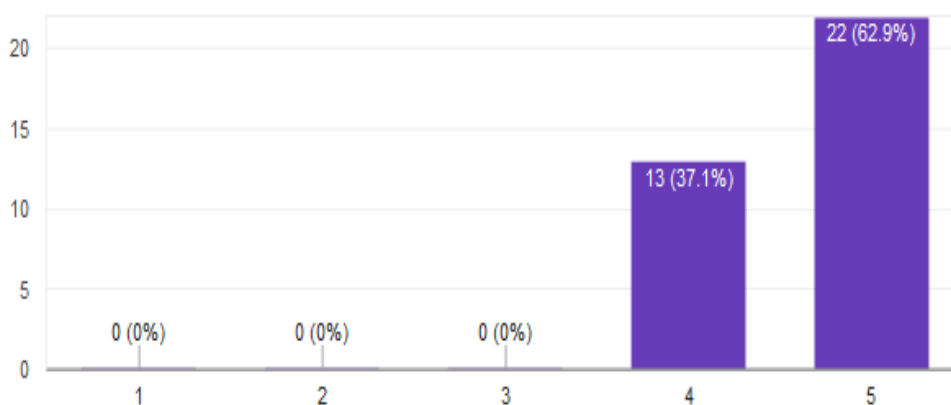


Figura 6.13: Resultado de encuesta después del sistema Pregunta 4

Fuente: El Autor

Facilidad de conseguir forma de contacto con el médico.

34 responses

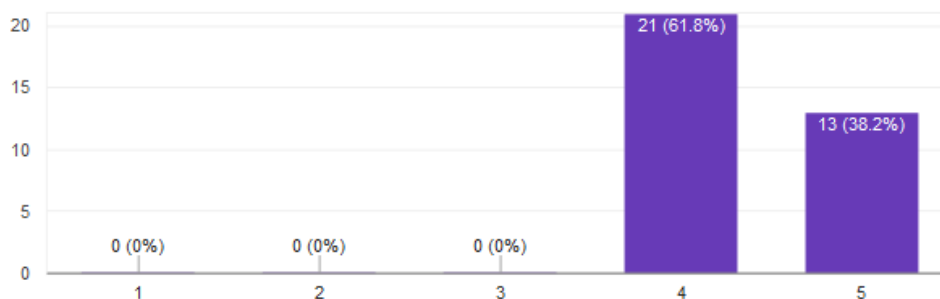


Figura 6.14: Resultado de encuesta después del sistema–Pregunta 5

Fuente: El Autor

Cuántas veces cada tres meses, por lo general, ha perdido el documento para solicitar exámenes médicos?

35 responses

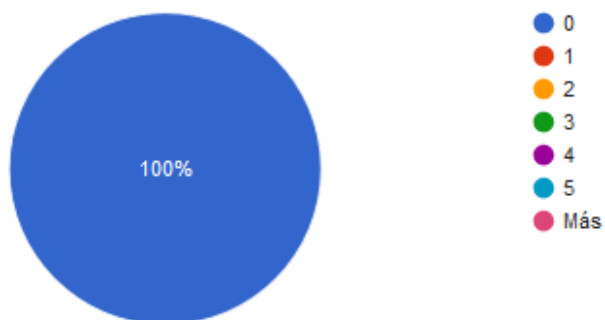


Figura 6.15: Resultado de encuesta después del sistema–Pregunta 6

Fuente: El Autor

Tiempo en consultar fecha de cita agendada, en caso de olvido.

35 responses

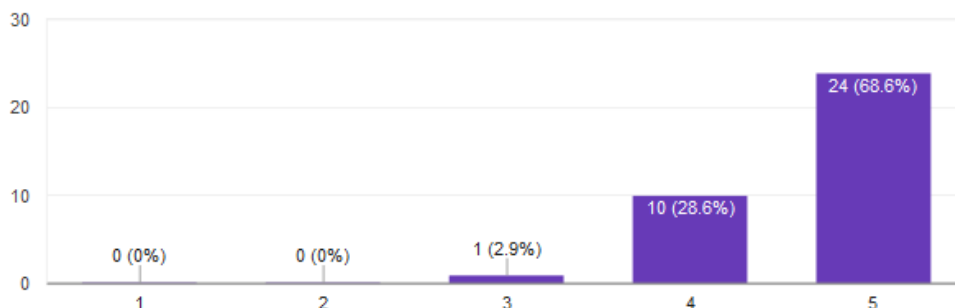


Figura 6.16: Resultado de encuesta después del sistema–Pregunta 7

Fuente: El Autor

Analizando los nuevos resultados, podemos observar que aparte de existir una mejora en el tiempo de respuesta al atender una cita médica por parte de los médicos; también se puede encontrar un mayor nivel de satisfacción del paciente en la forma de poder acceder a la información, ya sea del médico, historial médico, medicinas recetadas, exámenes de laboratorio y citas agendadas.

6.4 Análisis de cantidad de exámenes médicos de laboratorio alterados

En base a consultas realizadas a los miembros de la Sociedad de Trauma, se puede deducir que estos generan alrededor de 250 exámenes de laboratorio al mes, de los cuales unos 20 resultados han sido alterados por parte del paciente, ya sea, realizando resultados falsos casi idénticos a los documentos generados por los laboratorios, o tratando de modificar los valores mediante herramientas de edición de imágenes. El sistema trabaja

directamente con los laboratorios en la emisión de resultados. Por lo que la información de los resultados recibidos es completamente confiable, eliminando en su totalidad los resultados de laboratorio alterados.

CONCLUSIONES Y RECOMENDACIONES

Conclusiones:

En base al análisis realizado en las tareas diarias efectuadas por los miembros de la Sociedad de Trauma y en los resultados obtenidos, ya sea, por las encuestas realizadas a los pacientes y en el uso del sistema implementado por parte de los médicos; se puede concluir lo siguiente:

1. Conservar y acceder la información histórica médica o datos de gran relevancia de los pacientes para los médicos es una herramienta de mucha ayuda para poder brindar un diagnóstico más efectivo y más rápido.

2. Los pacientes de un médico valorizan mucho el acceso a herramientas que les ayuden a recordar la información brindada por su médico, siendo esto, medicamentos recetados junto con los detalles de su consumo, futuras citas agendadas, detalles de citas realizadas anteriormente, entre otros.
3. Existe manipulación en resultados de exámenes de laboratorio para que los pacientes puedan presentar información que le pueda favorecer, permitiendo que el médico realice un diagnóstico erróneo; por lo que se concluye que realmente necesario implementar técnicas de brindar seguridad a la información para que un médico pueda brindar un servicio de calidad a sus pacientes.
4. El uso de un sistema que tenga una interfaz intuitiva y bien diseñada ayuda al desempeño de las personas en el desarrollo de sus tareas diarias, siempre y cuando se muestre la información de manera adecuada para mejorar el análisis de un diagnóstico.
5. El acceso a retroalimentaciones alerta y ayuda de manera inmediata a la toma de decisiones para mejorar la satisfacción de un paciente.

Recomendaciones:

1. Incentivar el uso de la aplicación a otros capítulos de la Sociedad de Trauma, ayudaría a la generación de mayor información que puede ser de

utilidad para futuras investigaciones y estudios.

2. Desarrollar una aplicación móvil sería de gran utilidad para los pacientes, debido a que pueden tener su información de manera más directa, a parte se podrían generar los recordatorios en un dispositivo que el paciente tenga a su disposición todo el tiempo. También cabe mencionar que el acceso a los datos ya estaría implementado en el sistema REST.

BIBLIOGRAFÍA

- [1] Wikipedia, 'Modelo-vista-controlador', 2018. [Online]. Disponible: <https://es.wikipedia.org/wiki/Modelo%E2%80%93vista%E2%80%93controlador> [Accedido: 10-Sep-2018]
- [2] Expertiza, 'CSC/ECE 517 Fall 2014/ch1a 8 os', 2014, [Online]. Disponible: http://wiki.expertiza.ncsu.edu/index.php/CSC/ECE_517_Fall_2014/ch1a_8_os [Accedido: 10-Sep-2018]
- [3] MRBOOL, 'REST Architectural Elements and Constraints', 2016, [Online], Disponible: <http://mrbool.com/rest-architectural-elements-and-constraints/29339> [Accedido: 9-Sep-2018]
- [4] bbvaopen4u, 'REST API: What is it, and what are its advantages in project development?', 2016, [Online], Disponible: <https://bbvaopen4u.com/en/actualidad/rest-api-what-it-and-what-are-its-advantages-project-development> [Accedido: 9-Sep-2018]
- [5] Arquitectura Java, 'Introducción a JSON Web Token y la seguridad', 2017, [Online], Disponible: <https://www.arquitecturajava.com/introduccion-a-json-web-token/>, [Accedido: 10-Sep-2018]
- [6] IETF.org, 'JSON Web Token', 2015, [Online], Disponible: https://tools.ietf.org/html/rfc7519?utm_source=android-

[arsenal.com&utm_medium=referral&utm_campaign=2351,___](#) [Accedido: 12-Sep-2018]

[7] Certificationprocedures, 'Five Things You Should Know About Latest ISO 13485 Standard', 2018, [Online]. Disponible: <https://13485certificationprocedures.wordpress.com/tag/iso-13485-procedures/> [Accedido: 11-Sep-2018]