



UNIVERSIDAD TÉCNICA DE AMBATO

**FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E
INDUSTRIAL**

**CARRERA DE INGENIERÍA EN SISTEMAS
COMPUTACIONALES E INFORMÁTICOS**

TEMA:

**APLICACIÓN BASADA EN ARQUITECTURA ORIENTADA A SERVICIOS
PARA LA GESTIÓN DE LA INFORMACIÓN DE PACIENTES
HIPERTENSOS.**

Trabajo de Graduación. Modalidad: Proyecto de Investigación, presentado previo la obtención del título de Ingeniero en Sistemas Computacionales e Informáticos

LÍNEA DE INVESTIGACIÓN: Desarrollo de Software

AUTOR: Chisaguano Toapanta Edison Patricio

TUTOR: Ing. Dennis Chicaiza Mg.

Ambato - Ecuador

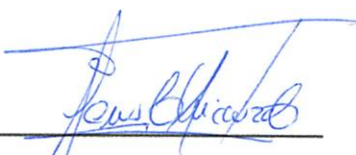
Octubre, 2018

CERTIFICACIÓN DEL TUTOR

En mi calidad de Tutor del Trabajo de Investigación sobre el Tema:

“APLICACIÓN BASADA EN ARQUITECTURA ORIENTADA A SERVICIOS PARA LA GESTIÓN DE LA INFORMACIÓN DE PACIENTES HIPERTENSOS”, del señor Chisaguano Toapanta Edison Patricio, estudiante de la Carrera de Ingeniería en Sistemas Computacionales e Informáticos, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, considero que el informe investigativo reúne los requisitos suficientes para que continúe con los trámites y consiguiente aprobación de conformidad con el Art. 16 del Capítulo II, del Reglamento de Graduación para Obtener el Título Terminal de Tercer Nivel de la Universidad técnica de Ambato

Ambato, octubre de 2018



Ing. Dennis Chicaiza Mg.

EL TUTOR

AUTORÍA DEL TRABAJO

El presente trabajo de investigación titulado: “APLICACIÓN BASADA EN ARQUITECTURA ORIENTADA A SERVICIOS PARA LA GESTIÓN DE LA INFORMACIÓN DE PACIENTES HIPERTENSOS”. Es absolutamente original, auténtico y personal, en tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, octubre de 2018



Edison Patricio Chisaguano Toapanta

CC: 1804367108

DERECHOS DE AUTOR

Autorizo a la Universidad Técnica de Ambato, para que haga uso de este Trabajo de Titulación como documento disponible para la lectura, consulta y proceso de investigación. Cedo los derechos de mi Trabajo de Titulación, con fines de difusión pública, además autorizo su reproducción dentro de las regulaciones de la Universidad.

Ambato, octubre de 2018

A handwritten signature in blue ink, appearing to read 'Edison Patricio Chisaguano Toapanta', written over a horizontal line.

Edison Patricio Chisaguano Toapanta

CC: 1804367108

APROBACIÓN DEL TRIBUNAL DE GRADO

La Comisión Calificadora del presente trabajo conformada por los señores docentes PhD. Julio Balarezo e Ing. Oswaldo Paredes Mg., revisó y aprobó el Informe Final del trabajo de graduación titulado “APLICACIÓN BASADA EN ARQUITECTURA ORIENTADA A SERVICIOS PARA LA GESTIÓN DE LA INFORMACIÓN DE PACIENTES HIPERTENSOS”, presentado por el señor Edison Patricio Chisaguano Toapanta de acuerdo al Art. 17 del Reglamento de Graduación para obtener el título Terminal de tercer nivel de la Universidad Técnica de Ambato.

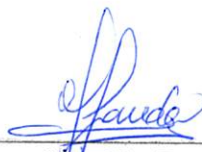


Ing. Pilar Urrutia Mg.

PRESIDENTE DEL TRIBUNAL



PhD. Julio Balarezo.
DOCENTE CALIFICADOR



Ing. Oswaldo Paredes Mg.
DOCENTE CALIFICADOR

DEDICATORIA

El presente trabajo quiero dedicar a Dios por haberme regalado la vida, la sabiduría por ser mi fortaleza en mis momentos de debilidad, así como también a mi padre Gaspar Chisaguano que desde el cielo me cuida y guía cada paso que doy, a mi madre Martha Toapanta que ha sido y será el pilar fundamental en mi convivir diario, para hoy cristalizar uno de mis objetivos de vida que me he fijado.

De igual manera a mis hermanos Edwin, Blanca y Jessica con quienes he compartido y aprendido tanto conocimiento como en acciones enfocadas a ser una persona de bien, con la satisfacción de ser un ejemplo ante ellos.

A mi enano querido Ariel con quien he compartido mi tiempo libre y he aprendido que alguien observa y sigue los pasos que se da.

A todos un Dios le pague de corazón.

Chisaguano Toapanta Edison Patricio

AGRADECIMIENTO

A Dios por la sabiduría otorgada para culminar uno de mis objetivos de vida, a mi madre, hermanos y sobrino por siempre confiar en mí y apoyarme en las cosas que me propongo realizarlas.

A los compañeros de clase que con el tiempo se convirtieron en amigos con los cuales se ha compartido muchos momentos en la etapa universitaria, y a uno en específico a quien lo estimo mucho por su amistad y ayuda obtenida de su parte.

A mi tutor de tesis, el Ing. Dennis Chicaiza por su asesoramiento durante el desarrollo del proyecto de investigación.

A los docentes de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial de la Universidad Técnica de Ambato por los conocimientos compartidos durante esta etapa de formación profesional.

Chisaguano Toapanta Edison Patricio

ÍNDICE

APROBACIÓN DEL TUTOR	ii
AUTORÍA	iii
AUTORÍA	iv
APROBACIÓN COMISIÓN CALIFICADORA	v
Dedicatoria	vi
Agradecimiento	vii
Introducción	xvi
CAPÍTULO 1 El problema	1
1.1 Tema de Investigación	1
1.2 Planteamiento del problema	1
1.3 Delimitación	2
1.4 Justificación	2
1.5 Objetivos	3
1.5.1 General	3
1.5.2 Específicos	3
CAPÍTULO 2 Marco Teórico	4
2.1 Antecedentes Investigativos	4
2.2 Fundamentación teórica	5
2.2.1 Hipertensión	5
2.2.1.1 Las señales más frecuentes de hipertensión	5
2.2.2 Service Oriented Architecture (SOA)	5
2.2.2.1 Arquitectura de un sistema SOA	6
2.2.3 Servicios Web	6
2.2.3.1 Ventajas de un SOA basado en Web Services	6
2.2.4 Metodologías de desarrollo	7

2.2.4.1	Metodología	7
2.2.4.2	Metodologías tradicionales	7
2.2.4.3	Metodología en Cascada	8
2.2.4.4	Metodologías ágiles	9
2.2.4.5	Metodología XP	10
2.2.4.6	Comparación de metodologías	12
2.2.5	Aplicaciones Informáticas	13
2.2.6	Base De Datos	13
2.2.7	Aplicaciones web	14
2.2.8	Google Charts	14
2.2.9	Visual Studio	14
2.2.10	Monitoreo	15
2.3	Propuesta de Solución	15
CAPÍTULO 3 Metodología		16
3.1	Modalidad Básica de la investigación	16
3.2	Población y muestra	16
3.3	Recolección de información	16
3.4	Procesamiento y análisis de datos	16
3.5	Desarrollo del Proyecto	17
CAPÍTULO 4 Desarrollo de la propuesta		18
4.1	Análisis	18
4.1.1	Ingeniería y Análisis del Sistema	18
4.1.1.1	Información sobre la hipertensión	18
4.1.1.2	Análisis sobre datos de pacientes hipertensos	24
4.1.1.3	Recolección de información sobre aplicaciones SOA	25
4.1.1.4	Determinar lenguaje de programación para la aplicación	27
4.1.2	Análisis de los requisitos del software	29
4.2	Diseño	31
4.2.1	Diseño de los procesos para la gestión de la información de pacientes hipertensos	31
4.2.1.1	Diagramas UML	31
4.2.1.2	Algoritmo para la gestión de la información de pacientes hipertensos	32
4.2.1.3	Casos de Uso	32
4.2.1.4	Diagrama de Actividades	34

4.2.1.5	Diagrama de despliegue	35
4.2.1.6	Diseño de Base de Datos	35
4.2.2	Diseño de Interfaces	39
4.3	Desarrollo	42
4.3.1	Codificación de la aplicación	43
4.4	Pruebas	56
4.4.1	Pruebas de caja blanca	56
4.4.2	Prueba de caja negra	60
4.4.3	Pruebas de rendimiento	68
CAPÍTULO 5 Conclusiones y Recomendaciones		69
5.1	Conclusiones	69
5.2	Recomendaciones	69
Bibliografía		71
ANEXOS		75

ÍNDICE DE TABLAS

1	Diferencias entre metodologías ágiles y tradicionales	12
2	Comparación metodología cascada y metodología XP	12
3	Clasificación de la presión arterial	21
4	Esquema de la recolección a practicar en la evaluación inicial de un paciente hipertenso	24
5	Comparación de lenguajes de programación	28
6	Descripción del actor administrador	34
7	Descripción del actor médico	34
8	Descripción del actor app móvil	34
9	Diccionario de datos.	37
10	Descripción Tabla Médico.	38
11	Descripción Tabla Paciente.	38
12	Descripción Tabla Medición.	38
13	Descripción Tabla Usuario.	39
14	Cuadro de pruebas: Método para consulta de datos de medición .	59
15	Complejidad ciclomática del método consulta medidas.	60
16	Cuadro de pruebas: Resultados - rendimiento	68
17	Cuadro de pruebas: Resultados - rendimiento (a)	68

ÍNDICE DE FIGURAS

1	Arquitectura de un sistema SOA	6
2	PAS - PAS	19
3	Ataque cardíaco	20
4	Recomendaciones para la medición de la PA	23
5	Plataforma .NET	27
6	Algoritmo para la gestión de información del paciente.	32
7	Diagrama de caso de uso administrador.	33
8	Diagrama de caso de uso médico.	33
9	Diagrama de caso de uso app móvil.	34
10	Diagrama de actividades.	35
11	Diagrama de despliegue.	35
12	Diagrama Entidad Relación.	36
13	Prototipo login.	39
14	Prototipo página maestra.	40
15	Prototipo módulo de médicos.	40
16	Prototipo módulo de pacientes.	41
17	Prototipo monitoreo PA	42
18	Esquema del sistema para gestión de información.	43
19	Nuevo proyecto.	44
20	Aplicación de servicios WCF.	44
21	Clase Servicio	45
22	Método para insertar una medida	46
23	Método para insertar un nuevo paciente	46
24	Clase interface	47
25	Descripción tipos de datos	48
26	Test servicio web	48
27	Test servicio web - métodos	49
28	Test servicio web - método listar médicos	50
29	Test servicio web - método medidas de los pacientes	50
30	Test servicio web - método nuevo médico	51

31	Test servicio web - médico insertado	51
32	Test servicio web - método para insertar un dato desde la app móvil	52
33	Fragmento de código de la página maestra de la aplicación	53
34	Identificación partes de gráfica hightcharts	54
35	Identificación partes de gráfica highstock	54
36	Código gráfica	56
37	Camino básico del método para consultar las medidas PA (a)	57
38	Camino básico del método para consultar las medidas PA (b)	58
39	Grafo de flujo del método para consultar las medidas PA	59
40	Inicio de sesión	60
41	Página rol administrador	61
42	Página rol médico	61
43	Página mantenimiento pacientes.	62
44	Notificación del sistema	62
45	Notificación al eliminar un registro	63
46	Búsqueda inteligente	63
47	Registro actualizado	64
48	Página de medidas	64
49	Gráfica estadística de la presión arterial	65
50	Manipulación de series	66
51	Filtrado de fechas	66
52	Opciones de exportación	67
53	Reporte estadístico gráfico de monitoreo presión arterial	67
54	Manual usuario - Ingreso al sistema	75
55	Manual usuario - pantalla inicio	76
56	Manual usuario - mantenimiento pacientes	77
57	Manual usuario - mantenimiento pacientes (a)	78
58	Manual usuario - mantenimiento pacientes (b)	78
59	Manual usuario - mantenimiento pacientes (c)	79
60	Manual usuario - mantenimiento pacientes (d)	79
61	Manual usuario - módulo medidas	80
62	Manual usuario - módulo medidas (a)	80
63	Manual usuario - módulo medidas (b)	81
64	Manual usuario - módulo medidas (c)	82
65	Manual usuario - módulo medidas (d)	83
66	Certificado - aprobación del proyecto	84

Resumen Ejecutivo

El presente trabajo de investigación denominado aplicación basada en arquitectura orientada a servicios para la gestión de la información de pacientes hipertensos se la realizo con la finalidad de aportar con una aplicación informática que ayude en el proceso de lectura sobre el estado de salud referente a la presión arterial, obteniendo así la suficiente información que colabore en el control de la frecuencia cardíaca de una persona, para lo cual se ha unido los recursos informáticos con la ciencia de la salud en especial con los médicos cardiólogos ya que son ellos quienes lo utilizan con sus pacientes.

Para este proyecto de investigación se trabaja con un aplicación móvil independiente el cual proporciona los datos de la presión arterial de una persona, para nosotros poderlos gestionar dicha información se desarrolló web services con los cuales almacenamos los datos respectivos de los pacientes en el servidor.

La aplicación informática permite gestionar la información de médicos, pacientes y sus respectivos datos de las medidas de la presión arterial de tal forma que con la información obtenida a través de los web services se genera una gráfica estadística con el fin de ser analizada y así poder obtener el diagnóstico sobre el estado de salud de un paciente.

Abstract

The present research work called application based on architecture oriented to services for the management of hypertensive patients information was done with the purpose of providing a computer application that helps in the reading process on the state of health regarding pressure blood, thus obtaining sufficient information to collaborate in the control of a person's heart rate, for which computer resources have been combined with health science, especially with cardiologists, since they are the ones who use it with their patients.

For this research project we work with an independent mobile application which provides the blood pressure data of a person, for us to be able to manage this information web services were developed with which we store the respective data of the patients on the server.

The computer application allows to manage the information of doctors, patients and their respective data of the blood pressure measurements in such a way that with the information obtained through the web services a statistical graph is generated in order to be analyzed and thus be able to get the diagnosis about the health status of a patient.

Glosario de términos y acrónimos

- PA: Presión arterial
- PAS: Presión arterial sistólica
- PAD: Presión arterial diastólica
- FC: Frecuencia cardíaca
- PM: Presión media
- SOA: Service Oriented Architecture - Arquitectura Orientada a Servicios

INTRODUCCIÓN

Las tecnologías de la información dentro de la ciencia de la salud actualmente desempeñan funciones de alto rendimiento en lo que si de automatizar procesos se trata, esta actividad con lleva a que el conocimiento de la informática para ciencias de la medicina, sea considerada como una herramienta facilitadora para determinados aspectos de procesos científicos como puede ser: explotación de datos, búsqueda, almacenamiento y gestión de la información de determinados pacientes, mediante el uso de las tecnologías de la información posibilita el almacenamiento masivo de datos médicos para poder ser tratados y gestionados de una forma sistemática agilizando el control y diagnóstico de posibles enfermedades en un paciente.

Uno de los objetivos principales de la tecnología de la información es prestar servicios a los profesionales de la salud a través de aplicaciones informáticas que ayuden así a mejorar en algo la calidad de vida de los pacientes por medio de un control anticipado y adecuado para la salud.

En este proyecto, se enfoca en la gestión de información y monitoreo de la presión arterial del paciente, en el cual por medio de gráficas estadísticas se muestran cada uno de los datos los cuales son la frecuencia cardíaca, presión sistólica, presión diastólica y presión media de un paciente con el propósito de ser analizada de forma visual y así se le pueda brindar una ayuda rápida y un tratamiento personalizado.

Capítulo I: Denominado “EL PROBLEMA”, se plantea el problema a investigar, la justificación de la investigación y los respectivos objetivos a llevar a cabo durante el desarrollo de la investigación.

Capítulo II: Contiene el “MARCO TEÓRICO”, en donde se establece los antecedentes investigativos referentes a la investigación, también contiene fundamentación teórica que sirve de soporte a la investigación y se establece la propuesta de solución del problema.

Capítulo III: Comprende todo lo relacionado a la “METODOLOGÍA”, el proceso de recolección de la información, el procesamiento y análisis de la

información se define las diferentes actividades necesarias para cumplir con los objetivos planteados.

Capítulo IV: Consta del “DESARROLLO DE LA PROPUESTA”, donde se detalla cada una de las actividades realizadas durante la investigación.

Capítulo V: Contiene las conclusiones y recomendaciones a las que se llegan en el transcurso del desarrollo del proyecto.

CAPÍTULO 1

El problema

1.1. Tema de Investigación

Aplicación basada en arquitectura orientada a servicios para la gestión de la información de pacientes hipertensos.

1.2. Planteamiento del problema

En la actualidad la integración de las Ciencias Médicas con el desarrollo de las Telecomunicaciones y la Informática Médica y su aplicación en las diferentes actividades del sector de la salud, hace posible conceptualizar el término de Telemedicina: Se define como “la distribución de servicios de salud, en el que la distancia es un factor crítico, donde los profesionales de la salud usan información y tecnología de comunicaciones para el intercambio de información válida para el diagnóstico, tratamiento y prevención de enfermedades o daños, investigación y evaluación; y para la educación continuada de los proveedores de salud pública, todo ello en interés del desarrollo de la salud del individuo y su comunidad”.

Aunque este concepto se ha definido así por la OMS desde 1998, existen evidencias que a partir de inicios de la centuria de 1900 se ha utilizado la medicina a distancia con el desarrollo de equipos para la transmisión de resultados de rayos X a través del telégrafo en Australia. Existen otros ejemplos de la utilización de medios de comunicación para la transmisión de información en diferentes actividades de atención de la salud en el mundo entero [1].

En países en vías de desarrollo como el Ecuador día a día los gobiernos nacionales y seccionales realizan grandes esfuerzos e inversiones públicas encaminadas a mejorar las condiciones de vida de sus habitantes, ven como el producto de estas inversiones cada vez llegan a ser menos representativas y efectivas considerando el crecimiento poblacional y el área de acción de sus políticas, entre unos de sus sectores que mayor incidencia tiene sobre la calidad de vida de sus habitantes se halla el sector de la salud [2]. En este sentido se puede observar que la tecnología informática en el Ecuador no es tratada de una forma óptima la cual ayude de manera sistemática con datos o información de personas hipertensas por medio de la red.

En la provincia de Tungurahua la hipertensión se asocia al sobrepeso y la diabetes mellitus debido al estilo de vida de sus habitantes y la alimentación rica en grasas y carbohidrato. Según información proporcionada por establecimientos de salud, el INEC ha establecido que la principal causa de muerte de la población de Tungurahua, es por insuficiencia cardíaca, con 228 casos que responden a una tasa de 47.2 muertes por cada 100.000 habitantes [3]. Tomando así en cuenta los datos estadísticos del INEC se ha analizado el desarrollo de una aplicación para la gestión y monitoreo con afán de proporcionar información sobre el estado de salud de pacientes hipertensos.

1.3. Delimitación

- Área Académica: Software.
- Línea de Investigación: Desarrollo de Software.
- Sublínea de investigación: Aplicaciones Web.
- Delimitación espacial: Facultad de Ingeniería en Sistemas, Electrónica e Industrial de la Universidad Técnica de Ambato.
- Delimitación temporal: La presente investigación se desarrolló en el periodo académico Marzo - Agosto 2018.

1.4. Justificación

La importancia de la presente investigación radica en implementar una aplicación para la gestión y monitoreo de datos de salud de pacientes hipertensos.

El proyecto de investigación tratado aporta de manera significativa ya que puesto en práctica; se logrará incrementar los recursos informáticos en beneficio del sector de la salud.

Los beneficiarios a cuales estará dirigida la aplicación basada en arquitectura orientada a servicios para la gestión de la información son pacientes, médicos, investigadores, estudiantes y personas naturales con el afán de proporcionar información sobre su estado de salud y simplifique así el diagnóstico o descarte de hipertensión.

El interés del presente proyecto se da porque en la actualidad se cuenta con hardware y software propietario de alto costo y gama que limitan a pacientes o galenos de salud adquirirlos por su valor.

El desarrollo de la aplicación de gestión y monitoreo pretende suprimir la limitante generada por los dispositivos comerciales que únicamente brindan herramientas de configuración y descarga de información para determinados sistemas operativos y haciendo recurrente el uso de licencias.

La aplicación permitirá al paciente y médico conocer la información detallada en tiempo real, utilizando dispositivos personales, sin importar características, haciendo necesario únicamente el acceso a la red y sin requerir software adicional. Además, permitirá conocer un diagnóstico parcial sin la necesidad hayan transcurrido 24 horas de estudio.

1.5. Objetivos

1.5.1. General

Desarrollar una aplicación basada en Arquitectura Orientada a Servicios para la gestión de la información de pacientes hipertensos.

1.5.2. Específicos

- Estudiar la Arquitectura Orientada a Servicios (SOA) - Service Oriented Architecture.
- Seleccionar la metodología apropiada para el desarrollo de la aplicación basada en Arquitectura Orientada a Servicios.
- Implementar la aplicación orientada a servicios para la gestión de la información de pacientes hipertensos de la provincia de Tungurahua.

CAPÍTULO 2

Marco Teórico

2.1. Antecedentes Investigativos

Los antecedentes que se ha encontrado son trabajos relacionados al presente tema de investigación el cual fue uno de ellos que trata de DISEÑO Y DESARROLLO DE UNA APLICACIÓN MÓVIL QUE FACILITE EL SEGUIMIENTO Y CONTROL DE PACIENTES HIPERTENSOS EN PANAMÁ. En la cual concluyen “En nuestro proyecto queremos ofrecer a los usuarios una herramienta que permita almacenar datos relevantes de su padecimiento de hipertensión arterial. Estos datos facilitan el seguimiento y control de la enfermedad y sobre todo le permite ver mediante un gráfico como se ha comportado las medidas obtenidas en un tiempo determinando. Este tipo de aplicaciones son una herramienta de control y sobre todo, permite educar a los usuarios sobre su enfermedad. . . .” [4].

Otro tema relacionado es DESARROLLO DE UN SISTEMA DE MONITOREO Y MEDICIÓN DEL RITMO CARDÍACO PARA PERSONAS HIPERTENSAS A TRAVÉS DE DISPOSITIVOS MÓVILES CON SISTEMA OPERATIVO ANDROID. Entre sus conclusiones esta “Con el dispositivo móvil cardiaco llamado ZEPHYR se ha demostrado que tiene una utilidad para el beneficio de las personas que tienen patologías cardíacas, de la cual puede ser usado mediante este sistema que se ha logrado interactuar con un dispositivo móvil telefónico que dará como resultado el control de la frecuencia cardíaca” [5].

Tema relacionado DESARROLLO DE UN MODELO DE GESTIÓN DE SERVICIOS DE TI PARA APLICACIONES DE TELEMEDICINA EN EL ECUADOR. Establece en sus conclusiones “La tecnología utilizada para la implementación de los servicios de Telemedicina para pacientes diabéticos e hipertensos en el Ecuador es secundaria y puede variar dependiendo del grado de tecnificación que un país o una empresa dispone para implementar servicios de Telemedicina.” [2].

2.2. Fundamentación teórica

2.2.1. Hipertensión

La hipertensión o presión arterial alta es uno de los trastornos causados por las preocupaciones permanentes, el abuso del alcohol y bebidas estimulantes como el café y el té, el hábito de fumar cigarrillo o tabaco, y el exceso de peso u obesidad. La hipertensión puede producir complicaciones graves al sistema cardiovascular, es decir al corazón, las arterias y las venas.

La presión arterial se puede describir como la fuerza o presión que ejerce la sangre sobre las arterias [6].

2.2.1.1. Las señales más frecuentes de hipertensión

Las señales más frecuentes de hipertensión o presión arterial alta son:

- Dolor de cabeza, especialmente en las mañanas.
- Irritabilidad o mal genio.
- Nerviosismo.
- Cansancio fácil.
- Desaliento.

Cuando la hipertensión es más avanzada se puede presentar dificultad para respirar, dolor en el pecho, palpitaciones e hinchazón en las piernas y los pies [6].

2.2.2. Service Oriented Architecture (SOA)

SOA es un estilo de Arquitectura de Software basado en la definición de servicios reutilizables, con interfaces públicas bien definidas, donde los proveedores y consumidores de servicios interactúan en forma desacoplada para realizar los procesos de negocio. Se basa en cuatro abstracciones básicas: servicios, application frontend, repositorio de servicios y bus de servicios. Un servicio consiste en una implementación que provee lógica de negocio y datos, un contrato de servicio, las restricciones para el consumidor, y una interfaz que expone físicamente la funcionalidad. Las application frontend consumen los servicios formando procesos de negocios. Un repositorio de servicios almacena los contratos de servicios y el bus de servicios interconecta las application frontend y los servicios [7].

2.2.2.1. Arquitectura de un sistema SOA

Al diseñar sistemas orientados a servicios es importante distinguir dos niveles:

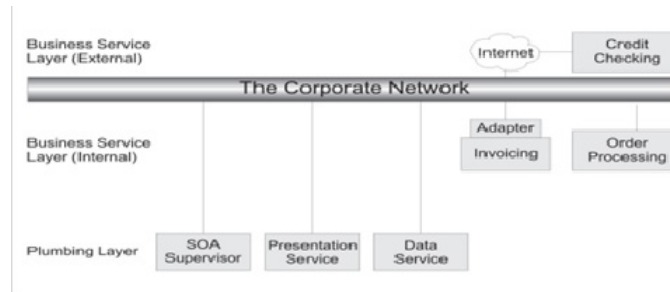


Figura 1: Arquitectura de un sistema SOA
Fuente: Revista Sistemas [8]

a) El nivel de Servicios de Negocio:

Contiene aquellos que componen los procesos de negocio, incluyendo la invocación a servicios web básicos, y la invocación de aplicaciones mediante adaptadores. Los servicios de negocio se subdividen en internos o externos, dependiendo de si son provistos por aplicaciones de la misma empresa o por aplicaciones externas.[8].

b) El nivel de Soporte Técnico (“plomería”):

Contiene los recursos computacionales y elementos de software que soportan los servicios de negocio. La separación de niveles permite hacer cambios al software de soporte técnico, en forma independiente de los cambios al software que realiza las funciones de los servicios de negocio[8].

2.2.3. Servicios Web

Los Servicios Web son una nueva generación de aplicaciones web, son sistemas de software diseñado para una interacción máquina a máquina a través de una red. Cuenta con una interfaz descrita en un formato procesable por una máquina. Los sistemas pueden interactuar con un servicio web mediante mensajes, normalmente se transmiten por medio de HTTP con una serialización XML, JSON, o texto simple en conjunto con otras normas relacionadas con la Web [9].

2.2.3.1. Ventajas de un SOA basado en Web Services

Un SOA basado en Web Services tiene las siguientes ventajas:

- Está basado en estándar (no propietario)
- Provee interoperabilidad de soluciones
- Soporta integración intra-organización, a través de la organización y entre organizaciones.

[9]

2.2.4. Metodologías de desarrollo

2.2.4.1. Metodología

Las metodologías de desarrollo de software son un conjunto de procedimientos, técnicas y ayudas a la documentación para el desarrollo de productos software [10]. La metodología también se define como la disciplina que indica y conlleva a que métodos y técnica utilizar en el desarrollo de un proyecto.

2.2.4.2. Metodologías tradicionales

Son una disciplina que tiene como base una gestión predictiva, es decir, que parte de los requisitos iniciales para configurar un plan adecuado usando los recursos y tiempo necesario, durante la fase de creación se comprueba si hay desviaciones, si las hay se definen las medidas a tomar y valorar cuales son las modificaciones que puede experimentar la planificación original. Por lo tanto define un conjunto de fases secuenciales en las que se indican las operaciones que se van a realizar, el tiempo que van a llevar [11].

Existen varias metodologías tradicionales para el desarrollo de software entre las cuales están:

- RUP (Rational Unified Procces)
- MSF (Microsoft Solution Framework)
- Win-Win Spiral Model
- Iconix
- Cascada.

2.2.4.3. Metodología en Cascada

Esta metodología es uno de los más básicos y ha servido como ejemplo para el desarrollo de los otros paradigmas del ciclo de vida. Este se basa en el lapso convencional de una ingeniería, su visión es muy clara: al desarrollar un software mediante una serie de fases secuenciales. Cada fase tiene metas ya establecidas, las cuales cuentan con actividades con una contribución al cumplimiento de dichas metas en cada una de las fases [12]. La metodología en cascada se caracteriza porque cada etapa se llevan a cabo una después de la otra que hay finalizado.

Características

- Metodología básica y más utilizada.
- Se rige a la ejecución de pasos secuenciales.
- Todas las fases cumplidas terminan un proyecto con éxito.
- Las fases continúan hasta cumplir con los objetivos planteados.
- Una fase no continúa sin haber terminado la anterior.

Ventajas de la metodología en cascada

- Gestiona el proyecto para su entrega a tiempo.
- Proporciona una gestión y planificación sencilla a los proyectos.
- Para poder continuar con el ciclo de las fases del paradigma se debe cumplir con los objetivos de una fase anterior.
- Cada fase genera resultados específicos.

Fases de la metodología en cascada

Ingeniería y Análisis del Sistema

El software forma parte de un sistema complejo que se encuentra dentro de una organización, en esta fase se obtiene todos los requisitos o elementos que serán necesarios para el completo funcionamiento al momento del desarrollo del sistema software [12].

Análisis de los requisitos del software

El ingeniero de software o el encargado de desarrollar un sistema debe comprender la información del software obtenida, así como la funciones o procesos que debe cumplir, la buena distribución de los recursos para así tener un excelente rendimiento [12].

Diseño

El diseño convierte los requisitos en una representación ya sea gráfica o estructural del software con calidad antes de que comience la codificación. Al momento de diseñar un software se debe enfocar en varios atributos; la organización de los datos, los detalles de los procesos, la arquitectura del software y una interfaz óptima [12].

Codificación

El diseño debe transformarse en una forma legible para la máquina. Si el diseño generado se encuentra bien detallado, al comenzar la codificación o al desarrollar el software, este se puede realizarse mecánicamente [12].

Prueba

Una vez obtenido el código o software comienza las pruebas del sistema. Las pruebas se centran en la lógica de programación y las funciones externas del software, realizando dichas pruebas se puede asegurar que las entradas definidas devuelven los resultados requeridos [12].

Mantenimiento

El mantenimiento realizado al software se da por motivo de errores, cambios del medio o requiriendo ampliaciones funcionales o del rendimiento [12].

2.2.4.4. Metodologías ágiles

Los métodos ágiles se originaron en el año 2001 por la inestabilidad del entorno técnico y porque el cliente a veces es incapaz de definir con exactitud los requisitos del proyecto de software. El término ágil se relaciona con la capacidad de adaptarse a los cambios de contexto y de especificaciones que ocurren durante el proceso de desarrollo [13]. Estas metodologías prácticamente son imprescindibles para cualquier tipo de proyectos expuestos a cambios concurrentemente.

Existen varias metodologías ágiles para el desarrollo de software entre las cuales están:

- Scrum
- XP - eXtreme Programming
- Crystal Methodologies
- Dynamic Systems Development Method
- Adaptive Software Development
- Feature -Driven Development
- Lean Development

2.2.4.5. Metodología XP

La programación extrema (XP) es una metodología ágil, se caracteriza porque los requerimientos del software se plantean como escenarios llamados historias de usuarios, los cuales se implementan como una serie de tareas que deben desarrollarse. Aplicando esta metodología los programadores trabajando en parejas hacen pruebas antes de escribir los programas [13].

Características

- Se realiza entregas pequeñas y frecuentes a los clientes.
- Tienen relación directa entre los clientes y el equipo de desarrollo.
- Su prioridad es el trabajo en equipo.
- Desarrollos previamente probados de las entregas, antes de su integración al sistema final.
- Se enfoca en el desarrollo de software antes que la documentación.

Ventajas de la metodología XP

- La gran ventaja de XP es su increíble capacidad de respuesta ante imprevistos, aunque por diseño es una metodología que no construye para el largo plazo y para la cual es difícil documentar.
- Bajo riesgo.
- Es una metodología flexible.

Fases de la Metodología XP son las siguientes:

Exploración

Es la fase en la que se define el alcance general del proyecto. En esta fase, el cliente define lo que necesita mediante la redacción de sencillas “historias de usuarios”. Los programadores estiman los tiempos de desarrollo en base a esta información. Debe quedar claro que las estimaciones realizadas en esta fase son primarias (ya que estarán basadas en datos de muy alto nivel), y podrían variar cuando se analicen más en detalle en cada iteración [14].

Planificación

La planificación es una fase corta, en la que el cliente, los gerentes y el grupo de desarrolladores acuerdan el orden en que deberán implementarse las historias de usuario, y, asociadas a éstas, las entregas. Típicamente esta fase consiste en una o varias reuniones grupales de planificación. El resultado de esta fase es un Plan de Entregas, o “Release Plan”, como se detallará en la sección “Reglas y Practicas” [14].

Iteraciones

Esta es la fase principal en el ciclo de desarrollo de XP. Las funcionalidades son desarrolladas en esta fase, generando al final de cada una un entregable funcional que implementa las historias de usuario asignadas a la iteración. Como las historias de usuario no tienen suficiente detalle como para permitir su análisis y desarrollo, al principio de cada iteración se realizan las tareas necesarias de análisis, recabando con el cliente todos los datos que sean necesarios. El cliente, por lo tanto, también debe participar activamente durante esta fase del ciclo [14].

Puesta en producción

Si bien al final de cada iteración se entregan módulos funcionales y sin errores, puede ser deseable por parte del cliente no poner el sistema en producción hasta tanto no se tenga la funcionalidad completa. En esta fase no se realizan más desarrollos funcionales, pero pueden ser necesarias tareas de ajuste[14].

2.2.4.6. Comparación de metodologías

Tabla 1: Diferencias entre metodologías ágiles y tradicionales

Elaborado por: Edison Chisaguano [15].

Metodologías Ágiles	Metodologías Tradicionales
Basadas en heurísticas provenientes de prácticas de producción de código	Basadas en normas provenientes de estándares seguidos por el entorno de desarrollo
Especialmente preparados para cambios durante el proyecto	Cierta resistencia a los cambios
Impuestas internamente (por el equipo)	Impuestas externamente
Proceso menos controlado, con pocos principios	Proceso mucho más controlado, con numerosas políticas/normas
No existe contrato tradicional o al menos es bastante flexible	Existe un contrato prefijado
El cliente es parte del equipo de desarrollo	El cliente interactúa con el equipo de desarrollo mediante reuniones
Grupos pequeños (<10 integrantes) y trabajando en el mismo sitio	Grupos grandes y posiblemente distribuidos
Pocos artefactos	Más artefactos
Pocos roles	Más roles
Menos énfasis en la arquitectura del software	La arquitectura del software es esencial y se expresa mediante modelos

Comparación de la metodología tradicional en cascada y la metodología ágil XP para la selección de la metodología apropiada para el proyecto denominado aplicación basada en Arquitectura Orientada a Servicios para la gestión de la información de pacientes hipertensos.

Tabla 2: Comparación metodología cascada y metodología XP

Elaborado por: Edison Chisaguano

Parámetro	Cascada	XP
Uso	Metodología básica y fácil implementación.	Metodología basada en varios principios y reglas para su implementación.
Documentación	Enfocada a la documentación. Detallada	Poca documentación. Ligera
Personal	Personal no experimentado	Personal experimentado
Especificación de requerimientos	Especificación de requerimientos al inicio del proyecto	Especificación de requerimientos en el trayecto de desarrollo del proyecto
Estilo	El estilo de desarrollo es anticipativo	Su estilo de desarrollo es adaptativo

2.2.5. Aplicaciones Informáticas

Según Bricopage afirma que las Aplicaciones Informáticas son aquellos programas que permiten la interacción entre usuario y computadora (comunicación), dando opción al usuario a elegir opciones y ejecutar acciones que el programa le ofrece. Existen innumerable número de tipos de aplicaciones.

Una aplicación, en final es un programa informático que permite a un usuario utilizar una computadora con un fin específico. Las aplicaciones son parte del software de una computadora, y suelen ejecutarse sobre el sistema operativo [16]. Existen programas denominados sistemas gestores de base de datos, abreviado SGBD (del inglés database management system o DBMS), que permiten almacenar y posteriormente acceder a los datos de forma rápida y estructurada. Las propiedades de estos DBMS, así como su utilización y administración, se estudian dentro del ámbito de la informática. Las aplicaciones más usuales son para la gestión de empresas e instituciones públicas; También son ampliamente utilizadas en entornos científicos con el objeto de almacenar la información experimental [16].

2.2.6. Base De Datos

Una base de datos o banco de datos es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso. En este sentido; una biblioteca puede considerarse una base de datos compuesta en su mayoría por documentos y textos impresos en papel e indexados para su consulta. Actualmente, y debido al desarrollo tecnológico de campos como la informática y la electrónica, la mayoría de las bases de datos están en formato digital, siendo este un componente electrónico, por tanto se ha desarrollado y se ofrece un amplio rango de soluciones al problema del almacenamiento de datos[17].

Existen programas denominados sistemas gestores de base de datos, abreviado SGBD (del inglés database management system o DBMS), que permiten almacenar y posteriormente acceder a los datos de forma rápida y estructurada. Las propiedades de estos DBMS, así como su utilización y administración, se estudian dentro del ámbito de la informática. Las aplicaciones más usuales son para la gestión de empresas e instituciones públicas; También son ampliamente utilizadas en entornos científicos con el objeto de almacenar la información experimental [17].

2.2.7. Aplicaciones web

Se denominan aplicaciones a aquellas aplicaciones cuya interfaz se construye a partir de páginas web. Las páginas web no son más que ficheros de texto en un formato estándar denominado HTML (HyperText Markup Language). Estos ficheros se almacenan en un servidor web al cual se accede utilizando el protocolo HTTP (HyperText Transfer Protocol), uno de los protocolos de internet. Para utilizar una aplicación web desde una máquina concreta, basta con tener instalado un navegador web [18].

2.2.8. Google Charts

Google Chart es una aplicación de Google para realizar estadísticas web, de fácil uso para desarrolladores de software web, usado en muchos campos como Google Analytics, se puede usar con diferentes formatos, Json, Javascript y plugins que se pueden integrar con varios lenguajes de programación [19]. Google Charts es una librería de Google la cual permite mostrar gráficas ya diseñadas en páginas web.

2.2.9. Visual Studio

Visual Studio es la interfaz de desarrollo de Microsoft. Se compone de un conjunto de herramientas que permiten a los desarrolladores crear aplicaciones para las plataformas .Net. Visual Studio se distribuye en varias ediciones: [20]

- Express.- Microsoft proporciona gratuitamente esta edición limitada, su objetivo es ayudar a la formación de desarrolladores, contando esta con funcionalidades básicas para la creación de proyectos.
- Professional.- Esta edición está más dirigida a desarrolladores con experiencia ya profesionales ya sean estos trabajen en equipo o de manera individual.
- Enterprise.- La siguiente versión lo utilizan equipos profesionales que trabajan en grupo y necesitan más interacción entre los miembros del equipo.
- Community.- Edición completa y sin paga es también una de las más utilizadas ya que incluye todas las herramientas de desarrollo multiplataforma para la creación de aplicaciones móviles Windows, iOS y android.

2.2.10. Monitoreo

La teoría de la planificación del desarrollo define el seguimiento o monitoreo como un ejercicio destinado a identificar de manera sistemática la calidad del desempeño de un sistema, subsistema o proceso a efecto de introducir los ajustes o cambios pertinentes y oportunos para el logro de sus resultados y efectos en el entorno. Así, el monitoreo permite analizar el avance y proponer acciones a tomar para lograr los objetivos; Identificar los éxitos o fracasos reales o potenciales lo antes posible y hacer ajustes oportunos a la ejecución [21].

2.3. Propuesta de Solución

En este proyecto se plantea el desarrollo de una aplicación basada en arquitectura orientada a servicios que busca principalmente proporcionar información sobre el estado de salud y simplifique así el diagnóstico o descarte de hipertensión.

CAPÍTULO 3

Metodología

3.1. Modalidad Básica de la investigación

El presente proyecto se basará en las siguientes modalidades de investigación:

La modalidad bibliográfica porque utilizara fuentes como libros, documentos, artículos, revistas, etc. Por medio de esta modalidad se pretende obtener toda información relevante que ayude al desarrollo teórico de la investigación.

La modalidad aplicada se realizará al aplicar los conocimientos adquiridos a lo largo de la carrera universitaria

3.2. Población y muestra

La presente investigación por su característica no requiere población y muestra.

3.3. Recolección de información

Para la recolección de información se utilizará la técnica de investigación documental de tipo informativa, obteniendo información relevante de diferentes fuentes confiables, analizando y seleccionando la información de mayor relevancia la cual servirá de apoyo para el desarrollo de este proyecto.

3.4. Procesamiento y análisis de datos

Para el procesamiento y análisis de los datos se realizará las siguientes actividades:

- Recolección de datos mediante investigación documental.
- Análisis de la información obtenida.
- Tabulación de la información recogida.
- Interpretación de los resultados mediante gráficos y cuadros informativos.

3.5. Desarrollo del Proyecto

Analizar una metodología para llevar a cabo las siguientes actividades del proyecto

- Análisis del problema
- Buscar información sobre la hipertensión.
- Análisis sobre datos de pacientes hipertensos.
- Recolección de información sobre aplicaciones SOA
- Determinar lenguaje de programación para la aplicación.
- Diseño interfaces.
- Implementación de la aplicación.
- Pruebas
 - Caja blanca
 - Caja negra

CAPÍTULO 4

Desarrollo de la propuesta

En esta etapa se enfoca en el desarrollo del sistema para la gestión de información de personas hipertensas, para lo cual la metodología de desarrollo se analizó el modelo tradicional en cascada y el modelo ágil XP respectivamente siendo así elegida y conveniente para su aplicación en el presente proyecto la metodología en cascada.

4.1. Análisis

En el análisis se realizó la investigación para determinar las necesidades de los usuarios y especificaciones, requerimientos que el sistema debe cumplir.

4.1.1. Ingeniería y Análisis del Sistema

4.1.1.1. Información sobre la hipertensión

Hipertensión

Es la palabra que se le da a la presión arterial para describirla cuando esta esta alta.

Arterias

Son aquellas que conducen la sangre desde el corazón transportándolos así por medio de ellas hacia otros órganos del cuerpo humano.

Presión arterial

Es una medición de sangre impulsada o generada por el corazón que fluye por las paredes del sistema arterial.

La medición de la presión arterial es fácil de realizarlo tanto para el paciente como para cualquier persona en general siempre y cuando cuente con el aparato indicado y adecuado de medición, la presión arterial se la mide en milímetros de mercurio (mmHg) con la cual indica el nivel la presión arterial.

Ahora bien cabe recalcar que la presión arterial (PA) cuenta con dos componentes, denominados presión arterial sistólica (PAS) y presión arterial diastólica (PAD). La presión arterial sistólica (PAS) es aquella que viene determinado por el impulso cardíaco producido por las contracciones del ventrículo izquierdo es ahí cuando se la denomina alta. La presión arterial diastólica (PAD) es la baja que depende de las resistencias que oponen las arterias al paso de la sangre. Para las lecturas de la presión arterial por lo general se las dan como dos números, el primero que es superior se la llama presión arterial sistólica y al segundo inferior se la llama presión arterial diastólica un claro ejemplo es 120 sobre 80 (se la escribe de la siguiente manera 120/80 mmHg).

a) Sistólica b) Diastólica

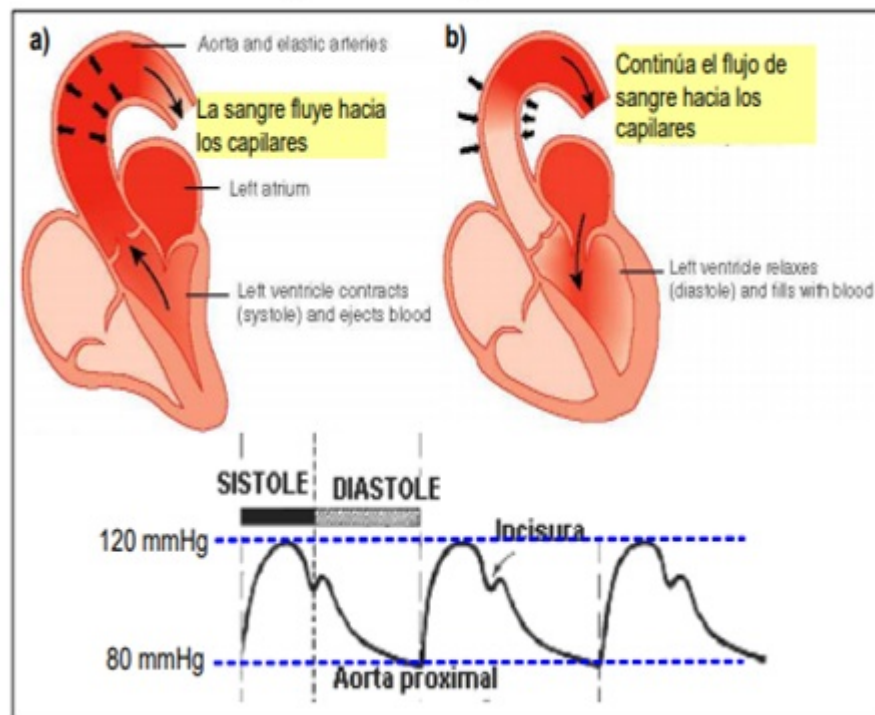


Figura 2: PAS - PAD
Fuente: U. Murcia [22]

La presión arterial alta es producida por el estrechamiento de unas pequeñas arterias llamadas arteriolas las cuales son quien regula el flujo de sangre en el organismo. Cuando las arteriolas se llegan a estrechar o contraer, el corazón tiende a esforzarse más de normal para bombear la sangre a través de aquel espacio que se ha reducido, es ahí donde la presión en los vasos sanguíneos aumenta.

Un dato importante sobre la hipertensión arterial a nivel médico, es que cuando un paciente tiene la presión arterial alta la llegan a conocer como muerte silenciosa

y sin ser tratada la presión arterial alta puede llegar al riesgo de que un paciente le ocurra un ataque al corazón o un accidente cerebrovascular (ataque cerebral).

Ataque cardíaco

La mayoría de los ataques cardíacos son provocados por un coágulo que bloquea una de las arterias coronarias. Las arterias coronarias llevan sangre y oxígeno al corazón. Si el flujo sanguíneo se bloquea el corazón sufre por la falta de oxígeno y las células cardíacas mueren. El término médico para esto es infarto de miocardio [23].

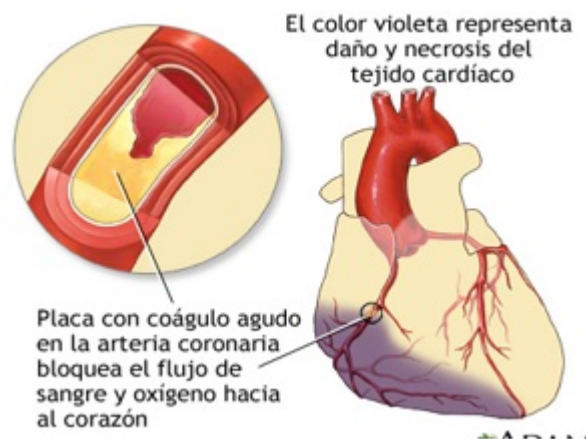


Figura 3: Ataque cardíaco
Fuente: MedinePlus [23]

Una de las causas del ataque cardíaco es la sustancia llamada placa la cual se acumula en las paredes de las arterias coronarias, esta placa se compone de colesterol y otras células. La causa de un ataque cardíaco no siempre se da a conocer ya que este puede ocurrir:

- Cuando la persona este descansando o dormido
- Después de haber realizado alguna actividad física.
- En un estrés emocional o físico.

Ataque cerebral

La Enfermedad Cerebro Vascular (ECV) es una patología que afecta a los vasos sanguíneos que suministran sangre al cerebro. A esta enfermedad también se le conoce como ictus, apoplejía, infarto cerebral, ataque cerebral, embolia o

trombosis cerebral. Según la Organización Mundial de la Salud (OMS) la ECV se define como el desarrollo rápido de signos clínicos de disturbios de la función cerebral o global con síntomas que persisten 24 horas o más, o que llevan a la muerte con ninguna otra causa evidente que el origen vascular [24].

La presión arterial es un parámetro variable

Es decir que nunca es constante ni a lo largo del día ni de la vida ya que este puede sufrir variantes puntuales con estímulos emocionales, esfuerzos físicos o mentales, también hay que tener en cuenta que la PA suele subir más en horas de trabajo por la presión que ello con lleva, y baja si el paciente está en reposo sin tensión alguna e inclusive más baja durante el lapso del sueño de la persona estas mediciones se las puede reconocer de manera sencilla con instrumentos especiales creados para monitorear la presión arterial durante las 24 horas.

Clasificación de la presión arterial por niveles

Tabla 3: Clasificación de la presión arterial

Elaborado por: Edison Chisaguano.

Grado	PAS		PAD
Óptima	<120		<80
Normal	120-129	y/o	80-85
Normal-alta	130-139	y/o	85-89
HTA de grado I	140-159	y/o	90-99
HTA de grado II	160-170	y/o	100-109
HTA de grado III	>180	y/o	>110
HTA sistólica aislada	>139		>89
HTA: hipertensión arterial, PAS: presión arterial sistólica, PAD: presión arterial diastólica.			

Por lo general al tomar la medida de la presión arterial del paciente se la puede pronosticar normal si es menor a 120/80 mmHg las veces que se le haya realizado la medición para el control de la presión arterial; por otro lado al realizar el control de la presión del paciente la medida es de 140/90 en adelante se le considera que la presión arterial es alta y por lo tanto el paciente es pronosticado con hipertensión; otros valores que se pueden dar son de 120/80 o más y no llega a los 140/90 el paciente es pronosticado como pre hipertensión.

Existen algunos factores que pueden afectar y causar hipertensión en las personas:

- Herencia.- esto ocurre cuando uno de los progenitores padece de hipertensión arterial esto hace que el niño/a padezca de igual manera con esta enfermedad.
- Consumo de sal.- al consumir en exceso sal de mesa también es un factor que incide y provoca que una persona sea propensa a ser hipertensa.
- Estrés.- el paciente debe procurar no permanecer preocupado ni nada por el estilo ya que el estrés cuenta con hormonas denominadas catecolaminas, dichas hormonas provocan una contracción de la musculatura de los vasos sanguíneos aumentando así la resistencia vascular y por lo tanto aumenta presión arterial en las personas.
- Sobrepeso.- otro factor que incide a la presión arterial alta es el sobrepeso ya que el estilo de vida que una persona lleva puede causar obesidad provocando así hipertensión arterial.
- Edad.- la persona al pasar los años e ir envejeciendo también vienen a ser propensos a tener la presión arterial alta por tal motivo deben llevar un control continuo de la presión arterial.

Síntomas y signos

El paciente puede presentar los síntomas y signos de hipertensión de forma muy leve e incluso pasar por desapercibido ya que no se las percibe claramente como otras enfermedades en algunos casos depende mucho del órgano afectado al corazón, arterias coronarias, riñón. Existen síntomas inespecíficos que se pueden presentar son mareos, ruidos cardíacos, cefaleas, palpitaciones entre otros más.

Tratamiento

El tratamiento que se le debe dar a un paciente detectado con hipertensión debe ser farmacológico ya que por medio del control se puede disminuir y controlar severas complicaciones de la hipertensión arterial de un paciente. Por otro lado también debe seguir un tratamiento no farmacológico como son la reducción de peso, no consumir alcohol, un consumo moderado de sodio, realizar ejercicio continuo evitar mantenerse estresado sin preocupaciones.

Medición de la presión arterial

Para la medición de la presión arterial se la debe efectuar de forma correcta a tal punto que se evite algún tipo de error en la toma de la medición de un paciente y así no contar con datos erróneos para lo cual se debe tomar las siguientes consideraciones:

- El paciente de preferencia debe permanecer sentado con la espalda recostada sobre el respaldar de la silla y su brazo reposar sobre una superficie plana, el antebrazo girarlo a la altura del corazón, y las piernas sin cruzarlos con los pies sobre el piso.
- Tomar la presión después de unos minutos de reposo, no debe haber ropa entre la piel y el manguito para medir.
- Se sugiere de preferencia emplear un tensiómetro de columna de mercurio el cual debe estar calibrado y revisado periódicamente para que no haya ningún inconveniente de medición.
- En una consulta lo ideal es tomar la presión en ambos brazos para definir cuál de ellos se encuentra más elevado e ir lo anotando en su expediente médico.

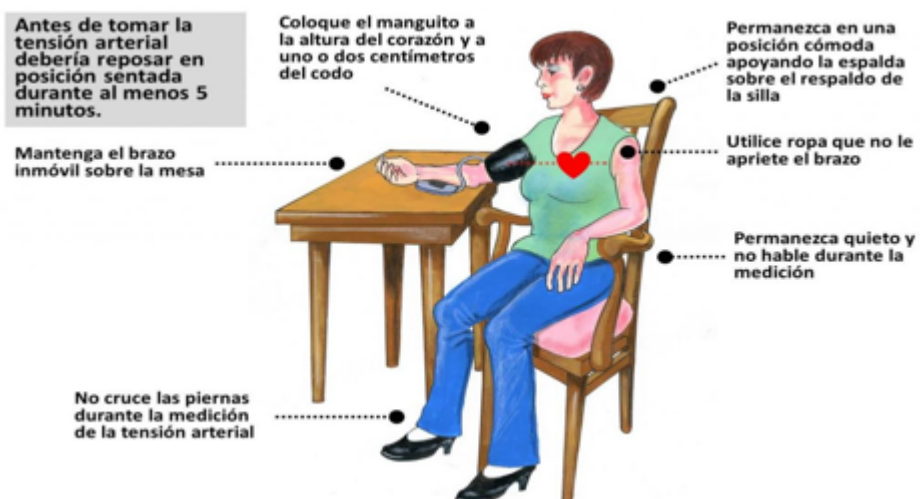


Figura 4: Recomendaciones para la medición de la PA

Fuente: CadisaúdeFerrol [25]

4.1.1.2. Análisis sobre datos de pacientes hipertensos

Para analizar información o datos de un paciente hipertenso se debe tener en cuenta que existen esquemas ya establecidos para llevar a cabo la evaluación clínica de un paciente por lo tanto un médico siempre debe registrar la duración del proceso hipertensivo, las causas de su descubrimiento, el nivel y los valores máximos de presión arterial que un paciente haya alcanzado en su medición y control arterial.

Los objetivos que se plantean en la evaluación de un paciente deben ser:

- Establecer un diagnóstico de hipertensión del paciente.
- Fijar si el paciente va tener beneficios con el tratamiento.
- Detectar la existencia de algunas otras enfermedades en el organismo del paciente.
- Identificar algún otro factor de riesgo vascular.

Tabla 4: Esquema de la recolección a practicar en la evaluación inicial de un paciente hipertenso

Elaborado por: IDIBAPS [26].

<ol style="list-style-type: none">1. Historia de la HTA: antigüedad, causas que motivaron su descubrimiento, cifras máximas de PA alcanzadas, tratamientos previos (eficacia y reacciones adversas)2. Consumo de fármacos o alimentos capaces de elevar las cifras de PA3. Sintomatología<ul style="list-style-type: none"><i>Neurológica:</i> cefalea, vértigos, inestabilidad, disminución de la libido, accidentes cerebro vasculares<i>Cardiovascular:</i> palpitaciones, disnea de esfuerzo, angina, infartos, edemas, claudicación intermitente<i>Renal:</i> glomerulonefritis, proteinuria, infecciones, cólicos, traumas renales, hematuria, poliuria4. Síntomas sugestivos de la posible existencia de una causa secundaria5. Historia familiar: poliquistosis, neurofibromatosis, neoplasia endocrina múltiple, defectos adrenales o gonadales, factores de riesgo de agregación familiar (HTA, diabetes, dislipemia, cardiopatía isquémica)6. Estilo de vida: dieta, ejercicio, consumos de tabaco y alcohol
--

Pulso arterial

Pulso arterial o frecuencia cardíaca son latidos que genera el corazón por minuto, el pulso de las personas es la expansión y retracción de la pared arterial como

la consecuencia al bombeo de la sangre en el corazón a causa de las variaciones de la presión arterial sistólica/diastólica con cada latido del corazón. Para poder tomar el pulso arterial se la tiene por lo general en las partes del cuerpo humano más visibles las arterias las cuales son las muñecas o el cuello.

Presión arterial media

Es una medida promedio de la medición de la presión arterial sistólica y la presión arterial diastólica la cual representa una presión estable o que resume en el cual fluiría la sangre si se movilizara de manera uniforme y no siguiendo el ciclo sístole diástole ya que el máximo valor que registra es durante la sístole (120mmHg) y va decayendo durante la diástole hacia un valor mínimo (80mmHg). Para calcular la presión arterial media se tiene una fórmula ya establecida la cual es: $PAM = (PAS + 2PAD) / 3$ con esta fórmula se puede sintetizar la presión que ejerce la sangre del corazón hacia adelante.

4.1.1.3. Recolección de información sobre aplicaciones SOA

SOA es una arquitectura de software que propone la construcción de aplicaciones mediante el ensamblado de bloques reusables, débilmente acoplados y altamente interoperables, cada uno de los cuales es representado como un servicio. Los mismos pueden encontrarse distribuidos y pertenecer potencialmente a diferentes propietarios. Es por ello que dicha arquitectura es utilizada para la integración de aplicaciones empresariales y comercio electrónico entre empresas [27].

SOA es una forma de tecnología que ayuda al desarrollo de aplicaciones con principios orientados a servicios web, por medio de SOA se pretende promover y apoyar toda la gestión de las reglas de negocio. SOA cabe recalcar que se base en principios los cuales son:

Encapsulación del servicio: Un desarrollador que crea un servicio por lo general es consolidado para que sean usados bajo SOA.

Servicio de acoplamiento flojo: Cada servicio mantiene una relación mínima entre dependencias.

Contrato de servicios: Los servicios son unidos a un acuerdo de comunicaciones con la cual se llega a describir y definir con mayor detalle la descripción del servicio.

Abstracción de servicio: En la abstracción se procede más allá del contrato de servicio es decir a la lógica que hay detrás de cada servicio.

Reutilización del servicio: Este principio trata de promover la reutilización con la lógica que se la divide en servicios.

Capacidad de servicio: Los servicios se los puede unir para formar otro servicio ya compuesto.

Autonomía de servicio: El servicio controla la lógica que encapsula.

Optimización del servicio: Igualdad de condiciones entre servicios de calidad alta y servicios de calidad baja.

Relevancia del servicio: El servicio debe contar con una funcionalidad significativo.

Factores que combinan SOA

- La simplicidad de los protocolos de acceso a objetos en los cuales se basan los servicios web.
- La complejidad de permitir interacciones de carácter autónomo e independiente.

Ventajas que proporciona SOA

- Permite ser reutilizado es decir que es flexible.
- Los clientes pueden consumir los servicios en diferentes aplicaciones a eso solo conoce como versatilidad.
- Optimizan el trabajo por medio de los datos y la coordinación que lleva.

SOA arquitectura que agiliza y facilita la implementación y creación de aplicaciones informáticas permitiendo así cumplir con los objetivos planteados de un proyecto también permite mantenerse enfocado en la regla de negocios. En dicha arquitectura los sistemas son altamente escalables ya que ayudan a las organizaciones un alto rendimiento al igual que una reducción de costos en sistemas de información, también mejora la flexibilidad en los procesos de negocios

4.1.1.4. Determinar lenguaje de programación para la aplicación

Para determinar el lenguaje hay que tener en cuenta que el proyecto se base en SOA ya que integra web services por ello se vio en la necesidad de seleccionar un lenguaje apropiado que permita realizar las tareas para el proyecto en cuestión.

Asp.Net

Asp.Net es una plataforma web la cual brinda los servicios necesarios para la compilación y ejecución de aplicaciones web y servicios web. Por la tanto en Asp.Net las aplicaciones se las desarrollan utilizando formularios web.

Asp.net forma parte de la plataforma .NET de ahí que windows forms y Asp.Net forms son las principales 2 herramientas con las cuales se pueden construir y desarrollar interfaces de usuarios en .Net. En la actualidad aun no son intercambiables ya que no existe una manera estándar en la que se diseñen y funcionen las interfaces tanto para aplicaciones de escritorio como aplicaciones web. Por lo tanto cada formulario cuenta con su posición relativa dentro de la familia de la tecnología en la plataforma .NET.

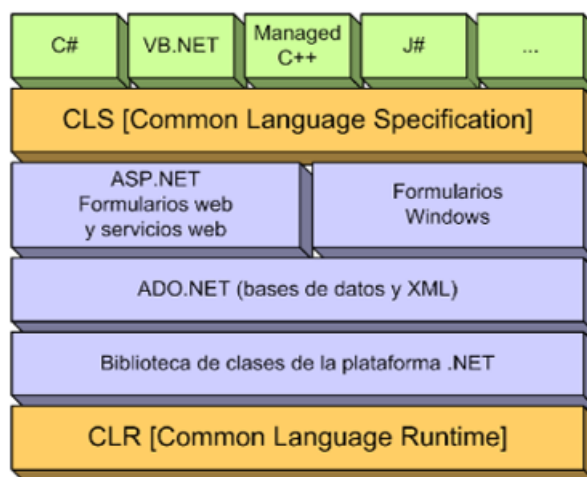


Figura 5: Plataforma .NET
Fuente: iKor Consulting [28]

La programación Asp.net está basada en el uso de controles y eventos de fácil manejo. Las paginas Asp.Net en vez de aceptar datos de entrada y generar su salida en HTML como suele suceder con Asp, implementan su funcionalidad en fragmentos de código que se ejecuta como respuestas a eventos asociados a los controles de interfaz con los que interactúan el usuario final. Esta forma de funcionar le proporciona a Asp.Net un mayor nivel de abstracción, requiere menos código permitiendo crear aplicaciones más mantenibles, modulares y legibles [28].

Comparación de lenguajes de programación

Tabla 5: Comparación de lenguajes de programación

Elaborado por: Edison Chisaguano [29].

Lenguaje	Características	Fortalezas	Debilidades
C#	Esta estandarizado por Microsoft como parte de su plataforma net.	Se desempeña de forma plena en los sistemas operativos Windows. Sintaxis más en comparación con C y C++. Posibilidad de realizar aplicaciones web, de escritorio y móviles	Requiere un mínimo de 4 GB para su instalación
JAVA	Multiplataforma	Al ser orientado a objetos permite su modulación. Permite la creación de aplicaciones de escritorio Tiene soporte a desarrollo de aplicaciones móviles y web.	Es un lenguaje interpretado así que es relativamente lento en comparación con otros lenguajes
PYTHON	Permite la creación de todo tipo de programas incluso sitios web. No requiere de compilación es un código interpretado	Libre y código fuente abierto. Lenguaje de propósito general. Multiplataforma. Orientado a objetos. Portable	Los lenguajes interpretados suelen ser relativamente lentos.

Lenguaje de programación C#

En la actualidad C# junto con java son lenguajes de programación más conocidos en internet, siendo así C# un lenguaje de alto nivel que pertenece al paquete de .Net junto con Visual Basic, C/C++ entre otros. Con C# se pueden escribir programas de escritorio como programas para la internet. C# es un lenguaje moderno y elegante, orientado a objetos creado por Microsoft con él se puede desarrollar aplicaciones seguras y solidas que se ejecutan en .Net su sintaxis es similar a la de java, C# es un lenguaje simple y flexible usado para crear aplicaciones cliente de Windows, aplicaciones cliente servidor, aplicaciones de base de datos, servicios web entre otras.

Características

- Admite múltiples paradigmas de programación
- Es multiplataforma.
- Su código es tratado como objetos.
- Sintaxis similar a java.
- Facilidad de uso y flexible

En este contexto, para el desarrollo de este proyecto se escogió como lenguaje de programación a C# por las características antes nombradas y basadas en el análisis realizado.

4.1.2. Análisis de los requisitos del software

En esta etapa se identificó y obtuvo información relevante al software sobre cuáles son los requisitos funcionales y no funcionales para el sistema de monitoreo de pacientes hipertensos.

Requisitos funcionales

Son aquellos que especifican claramente cuáles son las funciones o servicios que debe cumplir y hacer el sistema, para la aplicación de monitoreo de pacientes hipertensos se ha identificado los siguientes:

- Requisito funcional 1:
 - Autenticación de Usuarios: los usuarios deberán identificarse para acceder al sistema.
- Requisito funcional 2:
 - El sistema debe controlar el acceso de usuarios de acuerdo a roles.
- Requisito funcional 3:
 - Consultar Información: El sistema ofrecerá al usuario información de lo que este su alcance, por ejemplo consultar las medidas tomadas por el tensiómetro.
- Requisito funcional 4:
 - Reportes gráficos donde se pueda observar el comportamiento de las medidas de un paciente.

- Requisito funcional 5:
 - Registrar pacientes y médicos: El sistema permitirá al usuario registrar pacientes y médicos proporcionando datos como: Paciente (CI, nombre, apellido, edad, peso, estatura, dirección, teléfono) Médicos (CI, nombre, apellido, especialidad, dirección, teléfono).
- Requisito funcional 6:
 - Agregar, Modificar y Eliminar registros. El sistema debe tener botones para realizar estas acciones y solo lo podrá hacer si es usuario se ha identificado como administrador.
- Requisito funcional 7:
 - El sistema debe tener controlado los campos de registros para evitar datos incorrectos.
- Requisito funcional 8:
 - Desempeño eficaz del sistema. El sistema debe tener un buen rendimiento durante los procesos que se realicen.

Requisitos no funcionales

Los requisitos no funcionales son aquellos que describen las restricciones de los servicios o funciones que realiza el sistema de igual forma se ha identificado los siguientes para el sistema en cuestión:

- Requisito de rendimiento:
 - Garantizar que el diseño de las consultas u otro proceso no afecte el desempeño de la base de datos.
- Seguridad:
 - Garantizar la confiabilidad, la seguridad y el desempeño del sistema informático a los diferentes usuarios. En este sentido la información almacenada o registros realizados podrán ser consultados y actualizados permanente y simultáneamente, sin que se afecte el tiempo de respuesta.
- Fiabilidad:
 - El sistema debe tener una interfaz de uso intuitiva y sencilla.
 - El diseño debe ser responsivo

- Disponibilidad:
 - La disponibilidad del sistema debe ser continua con un nivel de servicio para los usuarios de 7 días por 24 horas.
- Usabilidad:
 - El sistema debe cumplir con una facilidad de uso que el usuario final interactúe sin problema alguno con la aplicación web.
- Soporte y mantenimiento:
 - Para este requisito el sistema contara con un manual de usuario en el cual se especificara detallada a profundidad cual es la manera de proceder de los usuarios ante el sistema y así pueda ver la funcionalidad de la aplicación.

Nota: Metadatos.- el usuario que manipule la aplicación será el responsable tanto de la integridad como la calidad de la información que se vaya a ingresar a la base de datos mediante la aplicación. Es así el usuario es netamente responsable del buen uso y correcto funcionamiento del sistema al momento de analizar los datos de un usuario y mostrarlos en la aplicación.

4.2. Diseño

En la siguiente fase se realizó las respectivas representaciones gráficas identificando los actores o beneficiarios para los procedimientos sistemáticos recolectados de los requerimientos para el sistema en cuestión.

4.2.1. Diseño de los procesos para la gestión de la información de pacientes hipertensos

Para el diseño de los procesos nos basamos en los requerimientos funcionales del sistema ya que demuestran gráficamente a detalle la funcionalidad y comportamiento que debe tener la aplicación.

4.2.1.1. Diagramas UML

Para la comprensión de los requerimientos funcionales del sistema se optó por la aplicación de diagramas UML (Lenguaje de Modelado Unificado), la cual es una sucesión de normas y estándares gráficos con el objetivo de representar la funcionalidad de un software a través de un esquema fácil de comprender. Entre los más utilizados están: Diagramas de Casos de Uso, Diagrama de

Clases, Diagramas de Secuencia, Diagramas de Estado, Diagramas de Actividad y Diagrama de Componentes.

4.2.1.2. Algoritmo para la gestión de la información de pacientes hipertensos

La Figura 6 representa el algoritmo que utiliza para la gestión de información de los pacientes.

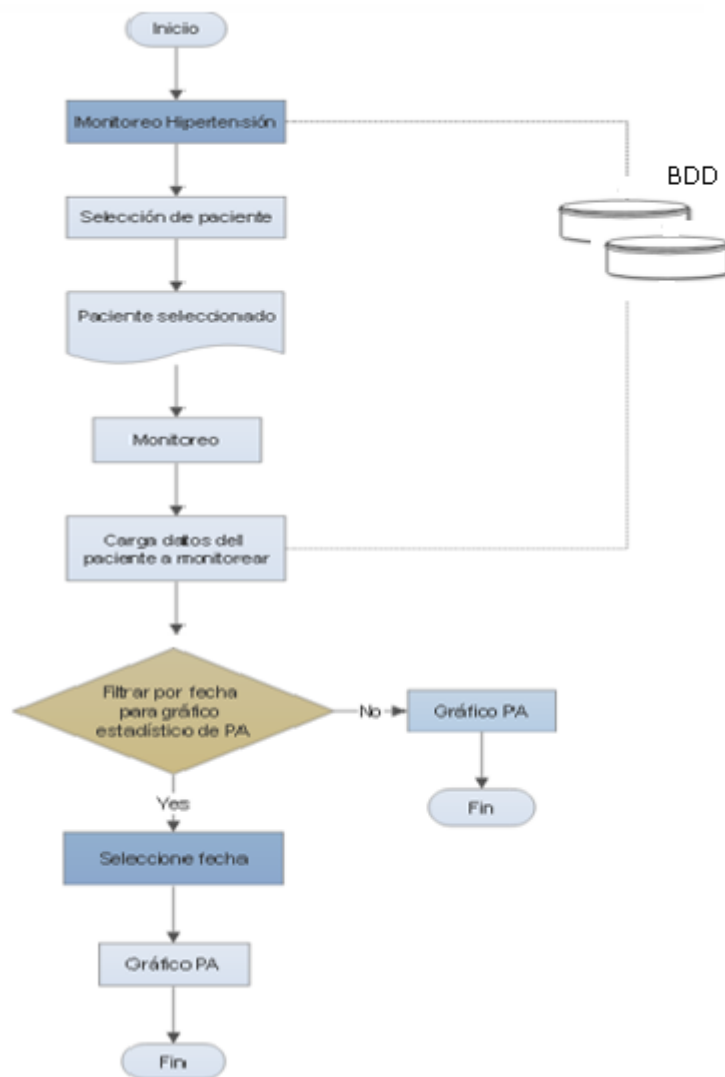


Figura 6: Algoritmo para la gestión de información del paciente.
Elaborado por: Edison Chisaguano.

4.2.1.3. Casos de Uso

Los casos de usos muestra a los actores (otros usuarios del sistema), los casos de uso (las situaciones que se producen cuando utilizan el sistema) y sus relaciones

o procesos principales de software.

La Figura 7 representa el caso de uso de la aplicación, actor administrador.

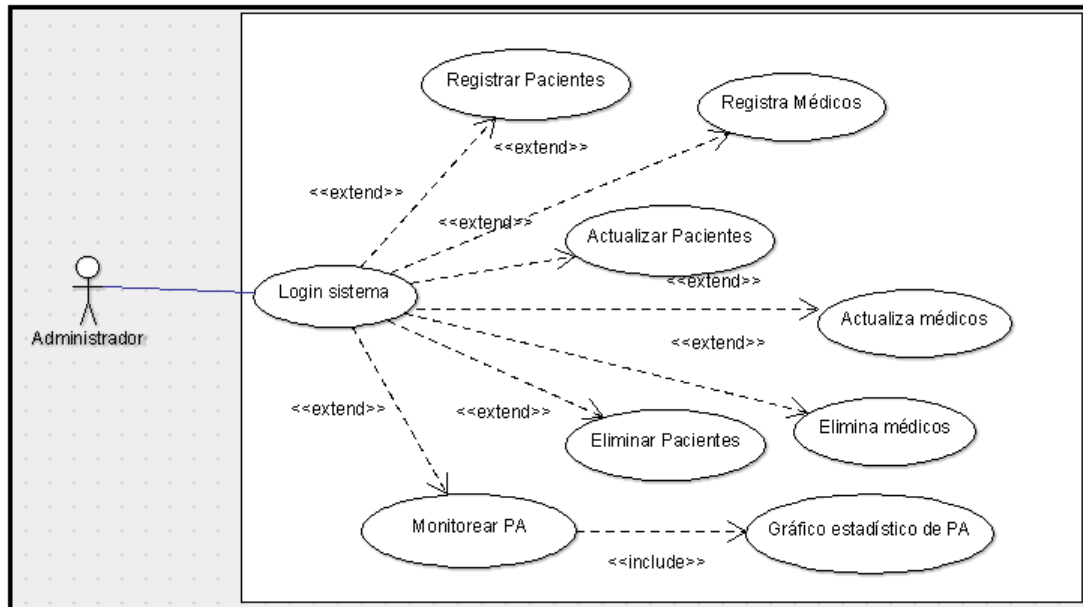


Figura 7: Diagrama de caso de uso administrador.
Elaborado por: Edison Chisaguano

La Figura 8 representa el caso de uso de la aplicación, actor médico.

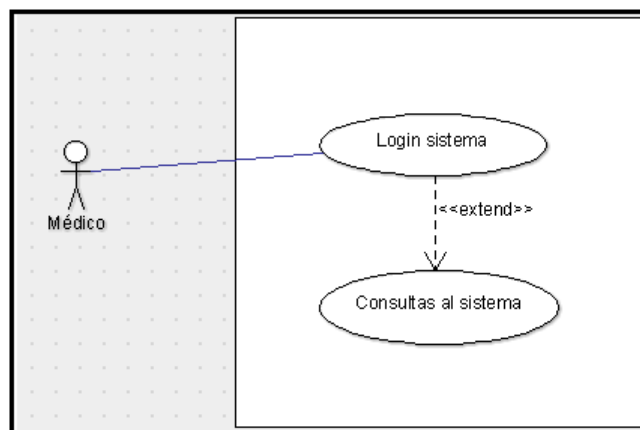


Figura 8: Diagrama de caso de uso médico.
Elaborado por: Edison Chisaguano

La Figura 9 representa el caso de uso de la aplicación, actor app móvil.

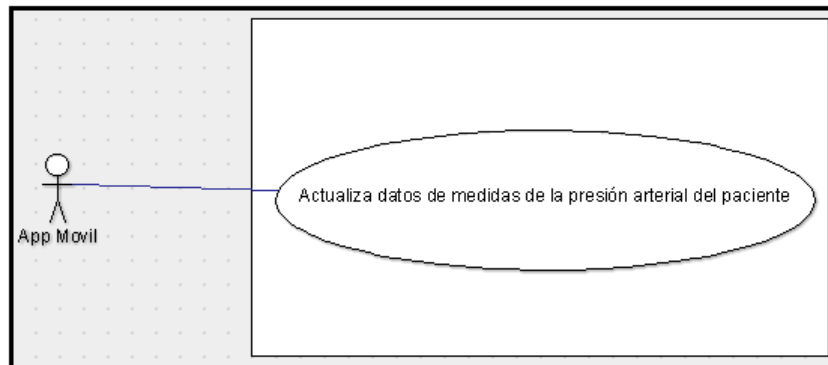


Figura 9: Diagrama de caso de uso app móvil.
Elaborado por: Edison Chisaguano

Descripción de los actores

Los actores identificados para la manipulación de la aplicación son:

Tabla 6: Descripción del actor administrador

Elaborado por: Edison Chisaguano

Actor 01:	Administrador
Descripción:	Administra el sistema, se encarga del mantenimiento de la información.

Tabla 7: Descripción del actor médico

Elaborado por: Edison Chisaguano

Actor 02:	Médico
Descripción:	Se encarga de generar consultas al sistema.

Tabla 8: Descripción del actor app móvil

Elaborado por: Edison Chisaguano

Actor 03:	App móvil
Descripción:	Se encarga de actualizar los datos de las medidas de los pacientes en el sistema.

4.2.1.4. Diagrama de Actividades

El diagrama de actividades muestra actividades, así como los cambios de una a otra actividad junto con los eventos que ocurren en ciertas partes del sistema. También mediante el diagrama de actividades se puede demostrar la lógica de

un algoritmo o el proceder del mismo mediante actividades las cuales siguen un camino lógico.

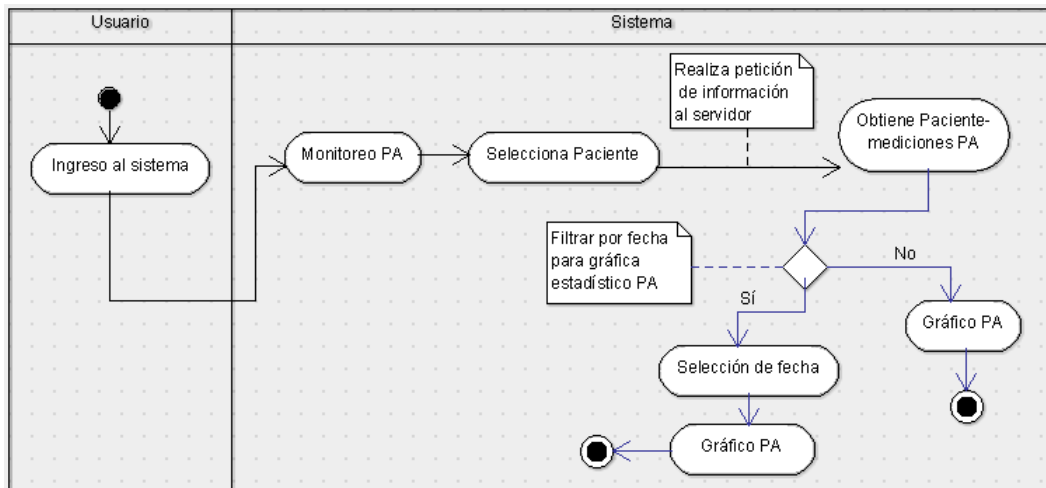


Figura 10: Diagrama de actividades.
Elaborado por: Edison Chisaguano

4.2.1.5. Diagrama de despliegue

Los diagramas de despliegue son aquellos que muestran la arquitectura física del sistema y la distribución de cada uno de sus componentes.

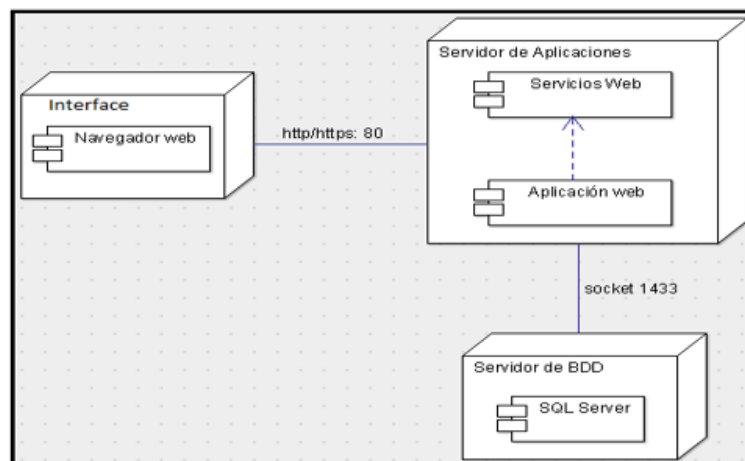


Figura 11: Diagrama de despliegue.
Elaborado por: Edison Chisaguano

4.2.1.6. Diseño de Base de Datos

Para llevar a cabo el diseño de la base de datos a usar en la aplicación de gestión de la información de pacientes hipertensos se lo realizó mediante el modelo

entidad relación con el cual ayuda representando gráficamente un esquema de la visión, percepción de un problema real. El modelo entidad relación prácticamente muestra las entidades y relaciones entre ellos; una entidad viene hacer un objeto real que se puede distinguir por sus diferentes características y las relaciones viene hacer las interacciones que pueden tener entre entidades. Figura 12 muestra el diseño de base de datos para el Sistema gestión de información de pacientes hipertensos.

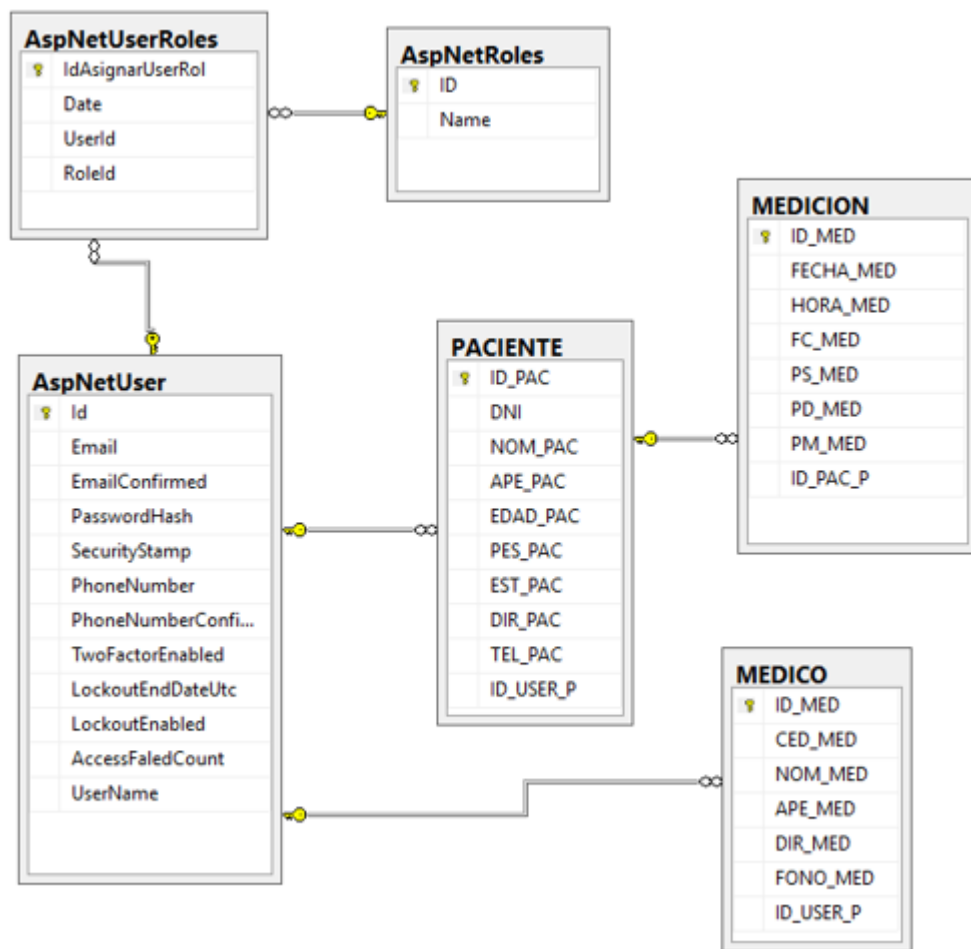


Figura 12: Diagrama Entidad Relación.
Elaborado por: Edison Chisaguano

Gestor de base de datos

Un gestor de base de datos es el que almacena y gestiona miles de datos atreves de un lenguaje definido para la manipulación de datos. Un gestor de base datos actúa como intermediario entre el usuario y los datos ya que es un conjunto de programas, que proporciona las herramientas necesarias para controlar u operar

las base de datos relacionales, siendo así su objetivo principal el de proporcionar un entorno eficiente, prácticamente permitiendo hacer todo en una base de datos. El gestor de base de datos para el presente proyecto de investigación es SQL Server Express Edition 2012 versión que no necesita un pago por la licencia y se lo puede utilizar sin inconvenientes, es aquella quién guardará toda la información para la gestión de pacientes hipertensos.

SQL Server

SQL Server es un sistema de gestión de base datos relacional, desarrollado por la empresa Microsoft, su código original fue desarrollado por Sybase a finales de 1980. SQL server proporciona un conjunto de herramientas para gestionar la información como son consultas, los procesos de análisis, formateo de datos entre sus características están la de proporciona seguridad a las aplicaciones, flexibilidad, confiabilidad y escalabilidad también soporta procedimientos almacenados.

Diccionario de datos

En el diccionario de datos se especificó a detalle las características de cada una de las tablas y atributos de la base de datos del presente proyecto.

Descripción de las tablas de la base de datos

Tabla 9: Diccionario de datos.

Elaborado por: Edison Chisaguano

Tabla	Descripción
USUARIO	Almacena información del usuario que acceder al sistema
MEDICO	Almacena información sobre datos personales de los médicos.
PACIENTE	Almacena información de datos personales de los pacientes.
MEDICION	Almacena información sobre los datos de las mediciones de pacientes emitidas por la aplicación móvil.

Descripción para la tabla médico.

Tabla 10: Descripción Tabla Médico.

Elaborado por: Edison Chisaguano

Nombre	Tipo de dato	Clave primaria	Obligatorio
ID_MED	int	Si	SI
CED_MED	char (10)	No	Si
NOM_MED	varchar (100)	No	Si
APE_MED	varchar (100)	No	Si
ESP_MED	varchar (100)	No	Si
DIR_MED	varchar (100)	No	Si
FONO_MED	varchar (10)	No	No
ID_USER_P	int	No	Si

Descripción para la tabla paciente.

Tabla 11: Descripción Tabla Paciente.

Elaborado por: Edison Chisaguano

Nombre	Tipo de dato	Clave primaria	Obligatorio
ID_PAC	int	Si	Si
DNI	chart (10)	No	Si
NOM_PAC	varchar (100)	No	Si
APE_PAC	varchar (100)	No	Si
EDAD_PAC	int	No	Si
PES_PAC	decimal (5,2)	No	Si
EST_PAC	decimal (5,2)	No	Si
DIR_PAC	varchar (20)	No	Si
TEL_PAC	int	No	No
ID_USER_P	int	No	Si

Descripción para la tabla medición.

Tabla 12: Descripción Tabla Medición.

Elaborado por: Edison Chisaguano

Nombre	Tipo de dato	Clave primaria	Obligatorio
ID_MED	varchar (5)	Si	Si
FECHA_MED	date	No	Si
HORA_MED	time (2)	No	Si
FC_MED	int	No	Si
PS_MED	int	No	Si
PD_MED	int	No	Si
PM_MED	int	No	Si
ID_PAC_P	varchar (10)	No	Si

Descripción para la tabla usuario.

Tabla 13: Descripción Tabla Usuario.

Elaborado por: Edison Chisaguano

Nombre	Tipo de dato	Clave primaria	Obligatorio
ID	int	Si	Si
LOGIN_ID	varchar (10)	No	Si
NOMBRES	varchar (100)	No	Si
PASS	varchar (100)	No	Si
ROL	varchar (20)	No	Si

4.2.2. Diseño de Interfaces

Para el diseño se realizan prototipos de las interfaces con el propósito de asimilar el entorno de la aplicación web permitiendo así validar el diseño de la interfaces de usuarios basándose en los requerimientos obtenidos en la etapa de análisis.

Inicio de sesión

En la imagen se muestra el diseño del login con el cual permite el acceso al sistema después de realizar el proceso de validación de los datos usuario y contraseña que estén almacenados en la base de datos, también dependerá de que rol tenga el usuario para poder acceder a los diferentes módulos de la aplicación.

INGRESO ALSISTEMA

Usuario

Contraseña

Ingresar

Figura 13: Prototipo login.
Elaborado por: Edison Chisaguano

Página maestra

Contiene un menú con los módulos que se mostraran en la aplicación y el usuario pueda acceder a cada uno de ellos dependiendo de su nivel de accesibilidad.

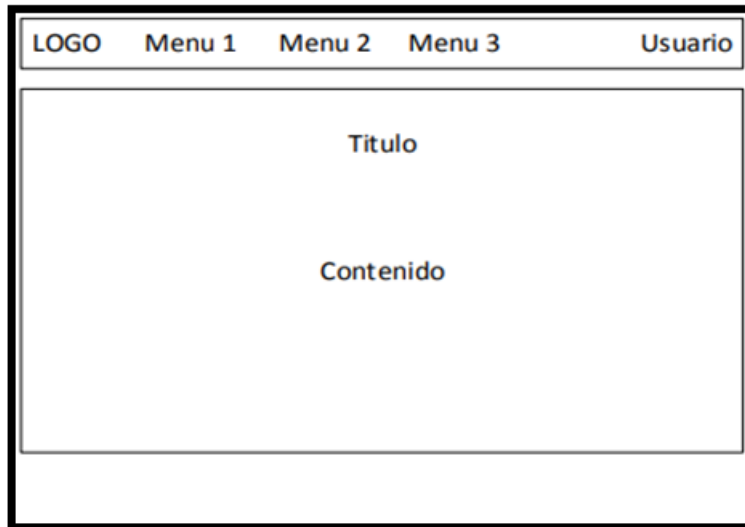


Figura 14: Prototipo página maestra.
Elaborado por: Edison Chisaguano

Módulo de médicos

En la siguiente imagen muestra el prototipo de diseño para el módulo de médicos con sus respectivos procesos de registrar, actualizar, eliminar un registro de la base de datos también contara un campo para realizar la búsqueda rápida de información.

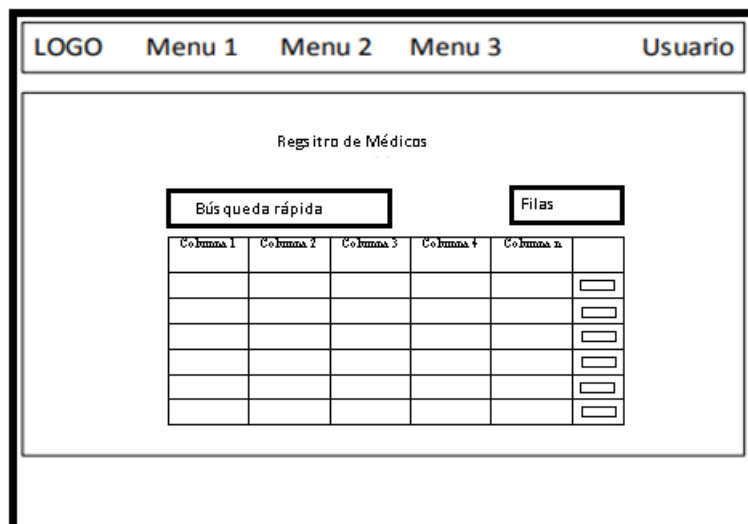


Figura 15: Prototipo módulo de médicos.
Elaborado por: Edison Chisaguano

Módulo de pacientes

Prototipo de diseño para el módulo de pacientes de igual forma con sus respectivos procesos de registrar, actualizar, eliminar un registro.

Columna 1	Columna 2	Columna 3	Columna 4	Columna n	
					<input type="button"/>
					<input type="button"/>
					<input type="button"/>
					<input type="button"/>
					<input type="button"/>

Figura 16: Prototipo módulo de pacientes.
Elaborado por: Edison Chisaguano

Monitoreo de la presión arterial

En la figura 15, muestra los procesos principales de aplicación, en la cual se presenta una tabla con todos los datos de pacientes almacenados en el servidor de base datos, ahí es donde el usuario seleccionara el paciente a monitorear la presión arterial, en la parte superior tiene un cuadro de búsqueda rápida de pacientes también cuenta con un botón en la lista para seleccionar un paciente y mostrar la información de medidas de presión arterial emitidas por la aplicación móvil a través del web service desarrollado para el proceso en cuestión, con dicha información se procede a realizar un gráfica dinámica y estadística con la información del paciente, en dicha gráfica se podrá seleccionar cual serie el usuario desea analizar y visualizarla en la presentación.

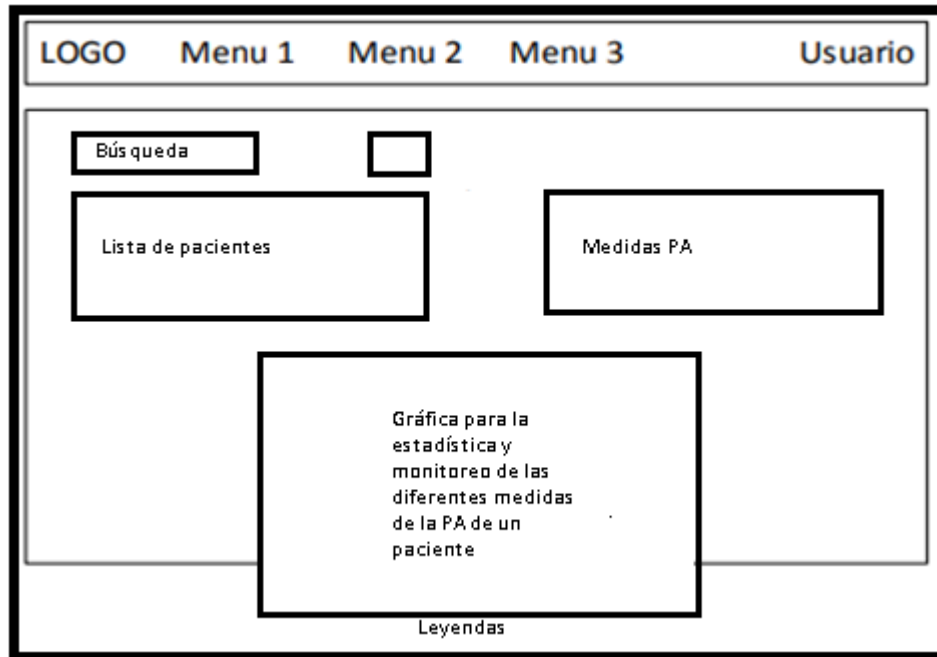


Figura 17: Prototipo monitoreo PA
Elaborado por: Edison Chisaguano

4.3. Desarrollo

En esta etapa para el desarrollo de la aplicación basada en la arquitectura orientada a servicios para la gestión de la información de pacientes hipertensos se la realizó con el IDE Visual Studio Community 2017 ya que es una edición completa y sin paga, y el lenguaje de programación C# con dicho lenguaje se implementó el código de servidor programando los procesos identificados en los requerimientos del usuario; el sistema de gestión de información de pacientes así misma también está conformada por un servidor web y un servidor de base datos. La aplicación web presenta la información de las medidas de la presión arterial de los pacientes en tiempo real emitidos por medio de una App móvil independiente, los datos de las medidas se obtienen mediante un web services que se implementó de igual manera para realizar el proceso de obtener las medidas de la presión arterial y almacenarlas en el servidor de base de datos y así poderlas gestionarlas en el sistema web; la aplicación web de gestión de información de los pacientes hipertensos y el respectivo web service están alojados en el servidor web, mientras que el servidor de base datos almacena la información de las medidas tomadas de la presión arterial de cada paciente.



Figura 18: Esquema del sistema para gestión de información.
Elaborado por: Edison Chisaguano

4.3.1. Codificación de la aplicación

En la etapa de codificación se realizó todo lo que es código fuente para el web service y la aplicación basada en arquitectura orientada a servicios para la gestión de información de pacientes hipertensos.

Creación del servicio web

Después de la creación de las tablas en la base de datos Sql Server se procede con el desarrollo de los servicios web para ello se utilizó wcf, los cuales fueron realizados con Visual Studio 2017.

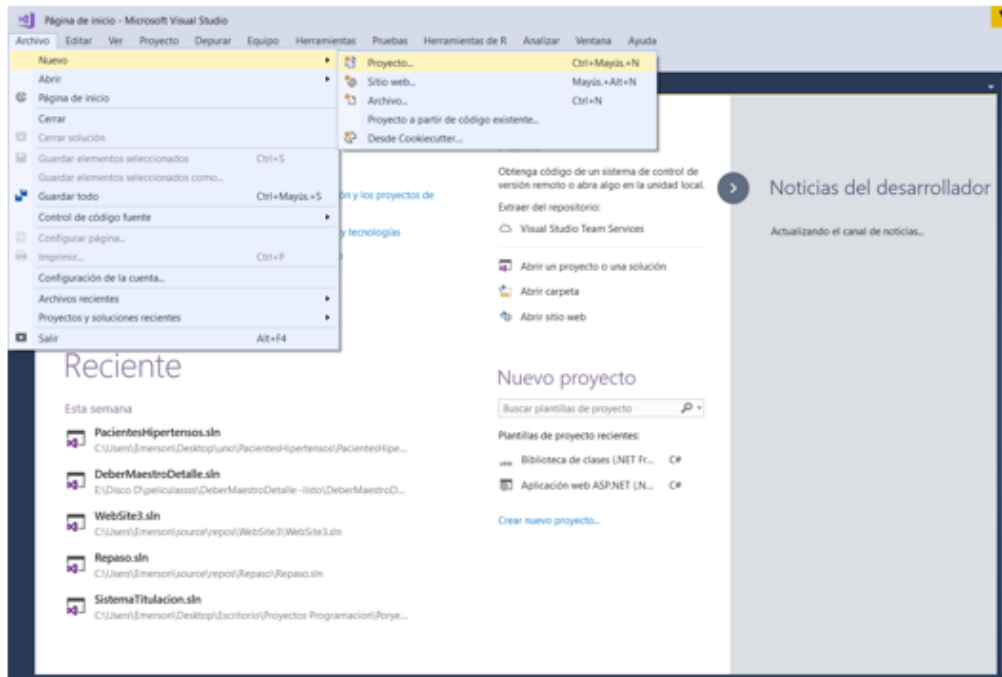


Figura 19: Nuevo proyecto.
Elaborado por: Edison Chisaguano

Al crear el proyecto seleccionamos Aplicación de servicios WCF

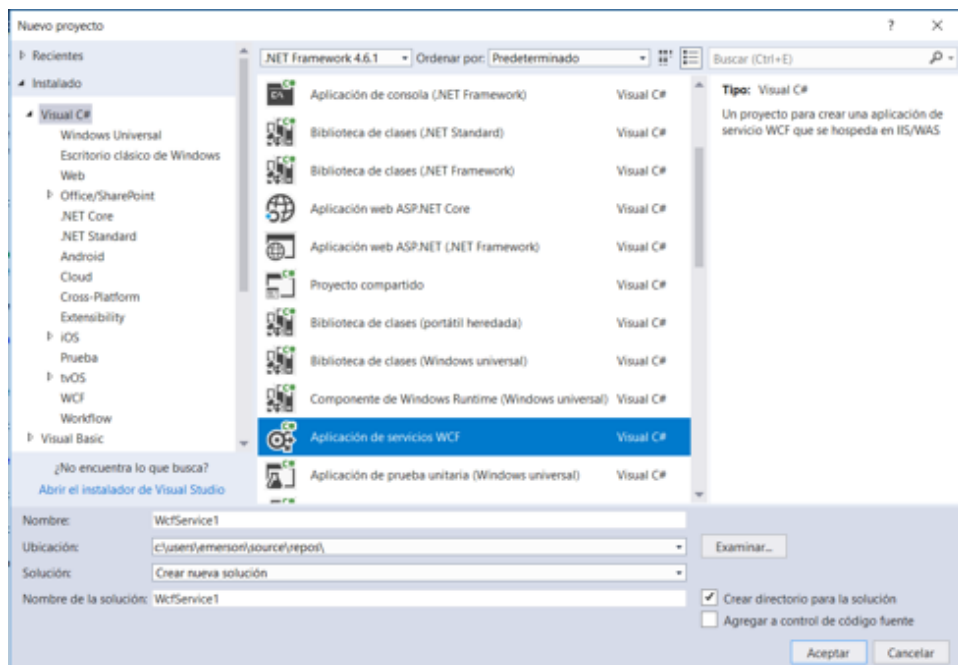


Figura 20: Aplicación de servicios WCF.
Elaborado por: Edison Chisaguano

A continuación se presenta fragmentos de código para la implementación del web service.

En la captura tenemos la clase que va recibir todas las peticiones al web service ahí están todos los métodos que el cliente va consumir con el servicio web dichos métodos están definidos en la interface llamada PacientesIService

```
namespace WcfServicios
{
    // NOTA: puede usar el comando "Rename" del menú "Refactori
    // NOTE: para iniciar el Cliente de prueba WCF para probar
    public class Service1 : PacientesIService
    {
        Conexion cnn = new Conexion();
        public int nuevoMedico(Medicos medicos)...
        public int editarMedico(Medicos medicos)...
        public int editarPaciente(Pacientes pacientes)...
        public int eliminarMedico(string id)...
        public int eliminarPaciente(string id)...
        public DataTable listMedico(string rango)...
        public DataTable listPacientes(string rango)...
        public Usuarios Login(string user, string pass)...
        public int nuevoPaciente(Pacientes pacientes)...
        public DataTable listPacientesMedidas(string rango)...
        public DataTable listMedidas(string id_paciente)...
        public int insertarMedidas(Medidas medida)...
    }
}
```

Figura 21: Clase Servicio
Elaborado por: Edison Chisaguano

Método para insertar una medida declarada en la clase servicio

```

public int insertarMedidas(Medidas medida)
{
    int result = 0;
    try
    {
        SqlCommand cmd = new SqlCommand("p_insert_medicion", cnn.abrir());
        cmd.CommandType = CommandType.StoredProcedure;
        cmd.Parameters.AddWithValue("@ID_MED", medida.ID_MED);
        cmd.Parameters.AddWithValue("@FECHA_MED", medida.FECHA_MED);
        cmd.Parameters.AddWithValue("@HORA_MED", medida.HORA_MED);
        cmd.Parameters.AddWithValue("@FC_MED", medida.FC_MED);
        cmd.Parameters.AddWithValue("@PS_MED", medida.PS_MED);
        cmd.Parameters.AddWithValue("@PD_MED", medida.PD_MED);
        cmd.Parameters.AddWithValue("@PM_MED", medida.PM_MED);
        cmd.Parameters.AddWithValue("@ID_PAC_P", medida.ID_PAC_P);

        result = cmd.ExecuteNonQuery();
        if (result == 0)
        {
            throw new Exception("No se ingreso la medicion, intentelo más tarde.");
        }
        cnn.cerrar();
    }
    catch (Exception ex)
    {
        new Exception(ex.Message);
    }
    return result;
}

```

Figura 22: Método para insertar una medida
Elaborado por: Edison Chisaguano

Método para insertar un nuevo paciente en la clase servicio

```

public int nuevoPaciente(Pacientes pacientes)
{
    int result = 0;
    try
    {
        SqlCommand cmd = new SqlCommand("p_insert_pacientes", cnn.abrir());
        cmd.CommandType = CommandType.StoredProcedure;
        cmd.Parameters.AddWithValue("@DNI", pacientes.DNI);
        cmd.Parameters.AddWithValue("@NOM_PAC", pacientes.NOM_PAC);
        cmd.Parameters.AddWithValue("@APE_PAC", pacientes.APE_PAC);
        cmd.Parameters.AddWithValue("@EDAD_PAC", pacientes.EDAD_PAC);
        cmd.Parameters.AddWithValue("@PES_PAC", pacientes.PES_PAC.Replace(',', '.').ToString());
        cmd.Parameters.AddWithValue("@EST_PAC", pacientes.EST_PAC.Replace(',', '.').ToString());
        cmd.Parameters.AddWithValue("@DIR_PAC", pacientes.DIR_PAC);
        cmd.Parameters.AddWithValue("@TEL_PAC", pacientes.TEL_PAC);
        cmd.Parameters.AddWithValue("@ID_USER_P", pacientes.ID_USER_P);

        result = cmd.ExecuteNonQuery();
        if (result == 0)
        {
            throw new Exception("No se ingreso el paciente, intentelo más tarde.");
        }
        cnn.cerrar();
    }
    catch (Exception ex)
    {
        //result = ex.Message;
        new Exception(ex.Message);
    }
    return result;
}

```

Figura 23: Método para insertar un nuevo paciente
Elaborado por: Edison Chisaguano

Definición de la interface

Se declara el atributo ServiceContract que es el que permite que se pueda utilizar esa interface como un contrato de servicio permitiendo que cada usuario que tenga acceso al servicio web utilice los métodos sin problema alguno.

```
[ServiceContract]
public interface IPacientesIService
{
    [OperationContract]
    Usuarios Login(string user, string pass);

    [OperationContract]
    DataTable listPacientes(string rango);
    [OperationContract]
    int nuevoPaciente(Pacientes pacientes);
    [OperationContract]
    int editarPaciente(Pacientes pacientes);
    [OperationContract]
    int eliminarPaciente(string id);

    [OperationContract]
    DataTable listMedico(string rango);
    [OperationContract]
    int nuevoMedico(Medicos medicos);
    [OperationContract]
    int editarMedico(Medicos medicos);
    [OperationContract]
    int eliminarMedico(string id);
}
```

Figura 24: Clase interface
Elaborado por: Edison Chisaguano

Los métodos que van hacer expuestos a través del servicio web las debemos declarar con OperationContract cada método tiene sus respectivos parámetros, etiquetado de igual forma con DataContract y a las propiedades se las identifica con DataMember para exponerlas en el servicio web.


```

[DataContract]
public class Medicos {
    string _ID_MED;
    string _CED_MED;
    string _NOM_MED;
    string _APE_MED;
    string _ESP_MED;
    string _DIR_MED;
    string _FONO_MED;
    string _ID_USER_P;
    [DataMember]
    public string ID_MED { get => _ID_MED; set => _ID_MED = value; }
    [DataMember]
    public string CED_MED { get => _CED_MED; set => _CED_MED = value; }
    [DataMember]
    public string NOM_MED { get => _NOM_MED; set => _NOM_MED = value; }
    [DataMember]
    public string APE_MED { get => _APE_MED; set => _APE_MED = value; }
    [DataMember]
    public string ESP_MED { get => _ESP_MED; set => _ESP_MED = value; }
    [DataMember]
    public string DIR_MED { get => _DIR_MED; set => _DIR_MED = value; }
    [DataMember]
    public string FONO_MED { get => _FONO_MED; set => _FONO_MED = value; }
    [DataMember]
    public string ID_USER_P { get => _ID_USER_P; set => _ID_USER_P = value; }
}

```

Figura 25: Descripción tipos de datos
Elaborado por: Edison Chisaguano

Luego de implementar cada uno de los respectivos métodos para el web service para la aplicación, pasamos a compilar para poderlos consumirlos en cualquier sistema

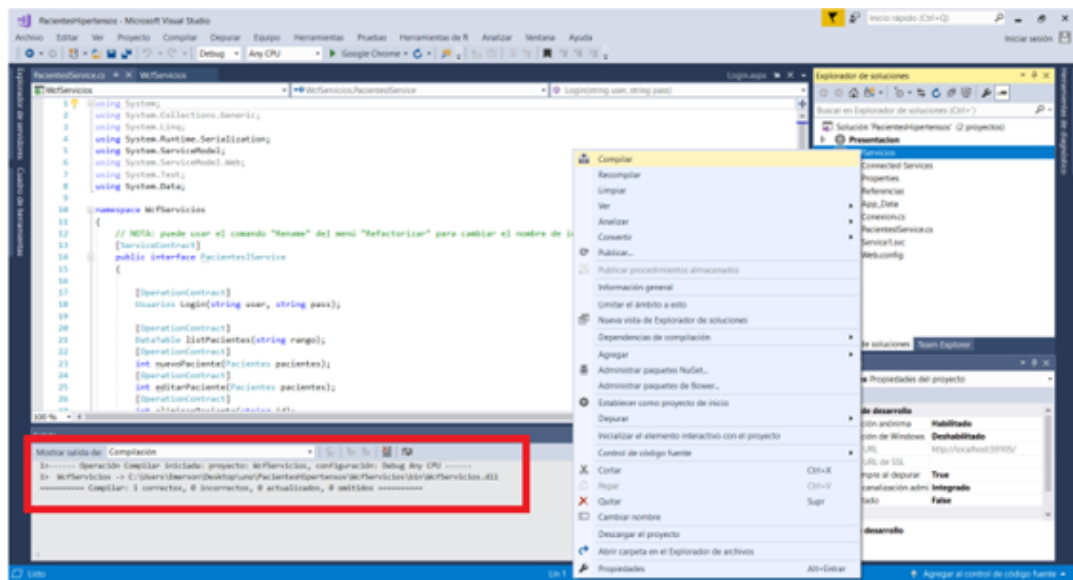


Figura 26: Test servicio web
Elaborado por: Edison Chisaguano

Test del web service

Para realizar la prueba de funcionamiento se utilizó la herramienta SoapUI ya que por medio de esta nos permite testear el servicio web sin la necesidad de ser

consumida en alguna aplicación.

Métodos reconocidos en SoapUI para la respectiva prueba .

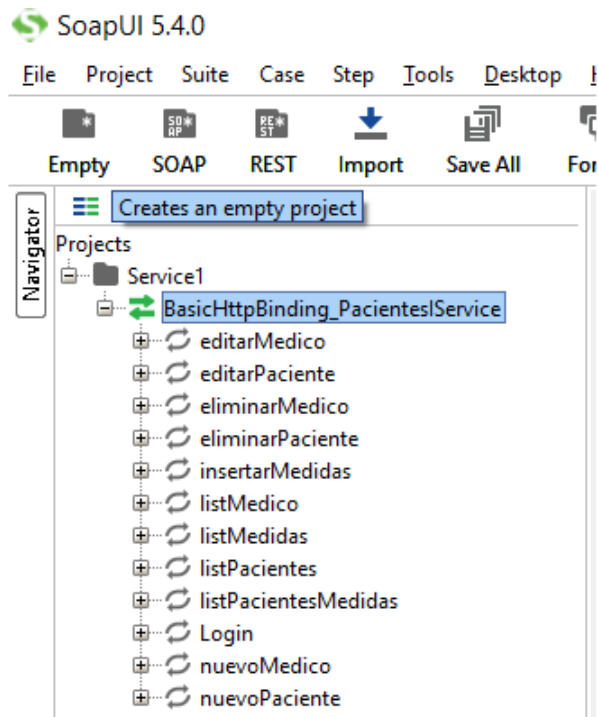


Figura 27: Test servicio web - métodos
Elaborado por: Edison Chisaguano

Método para listar los médicos.

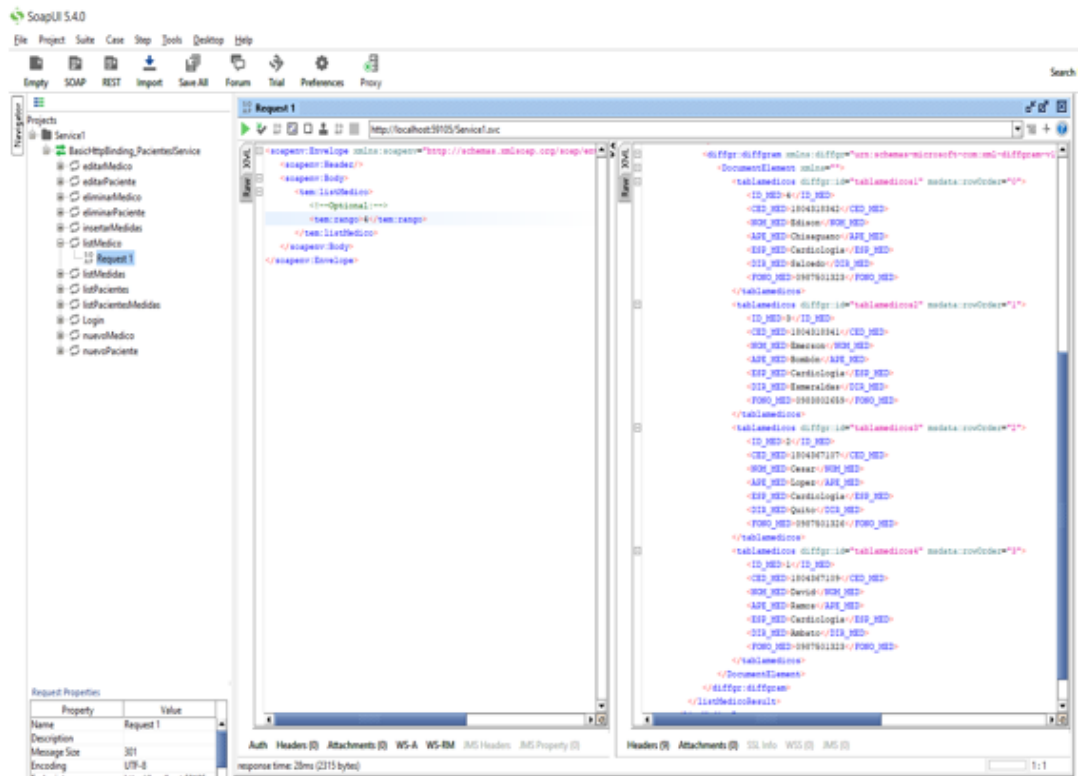


Figura 28: Test servicio web - método listar médicos
Elaborado por: Edison Chisaguano

Método para listar las medidas de los pacientes.

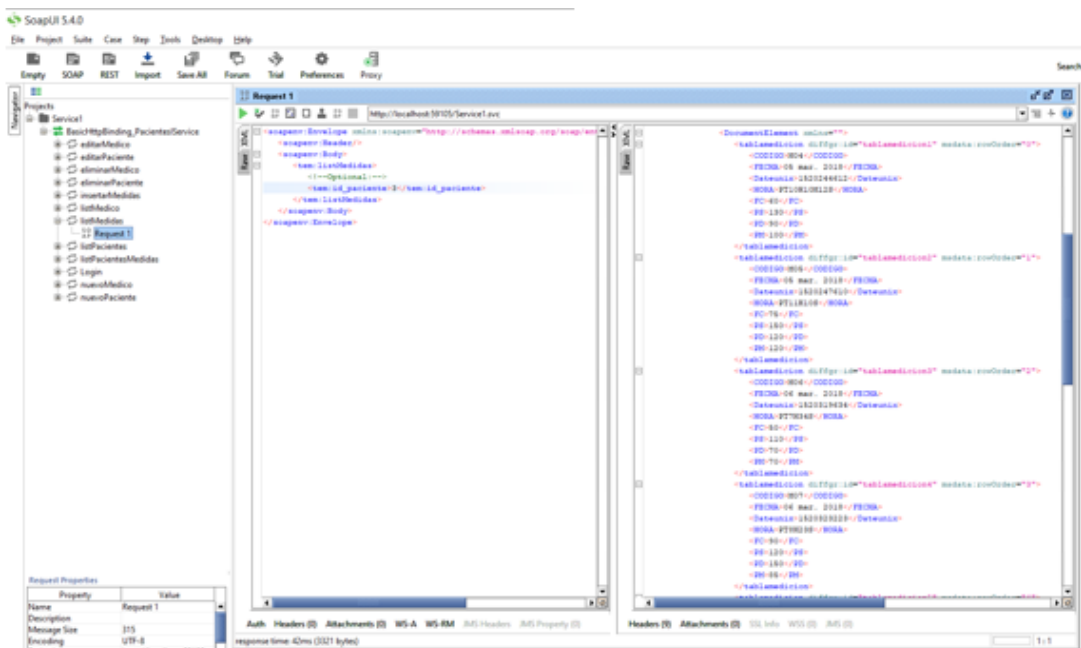


Figura 29: Test servicio web - método medidas de los pacientes
Elaborado por: Edison Chisaguano

Método para insertar un nuevo médico

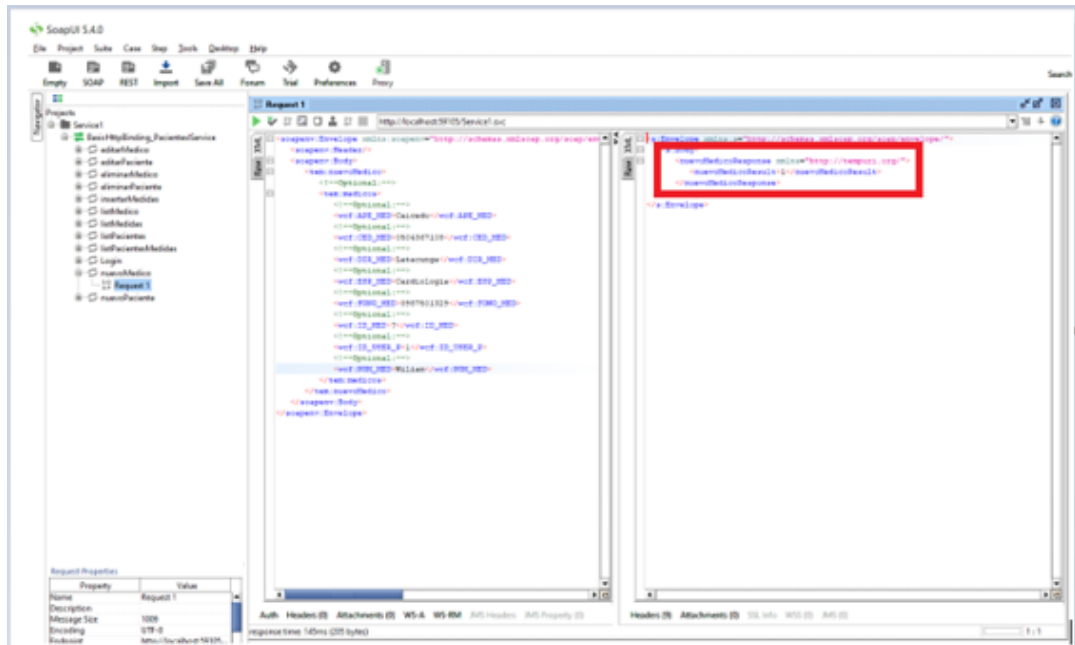


Figura 30: Test servicio web - método nuevo médico
Elaborado por: Edison Chisaguano

Devolviendo como resultado 1 si el servicio se ejecuta sin problema alguno, después pasamos a la base para ver el dato insertado

	ID_MED	CED_MED	NOM_M	APE_MED	ESP_MED	DIR_MED	FONO_MED	ID_USER_P
1	1	1804367109	David	Ramos	Cardiología	Ambato	0987501323	1
2	2	1804367107	Cesar	Lopez	Cardiología	Quito	0987501326	1
3	3	1804318341	Emerson	Bombón	Cardiología	Esmeraldas	0983802659	1
4	4	1804318342	Edison	Chisaguano	Cardiología	Salcedo	0987501323	1
5	5	0504367108	Wilian	Caicedo	Cardiología	Latacunga	0987501329	1

Figura 31: Test servicio web - médico insertado
Elaborado por: Edison Chisaguano

Insertando una medida

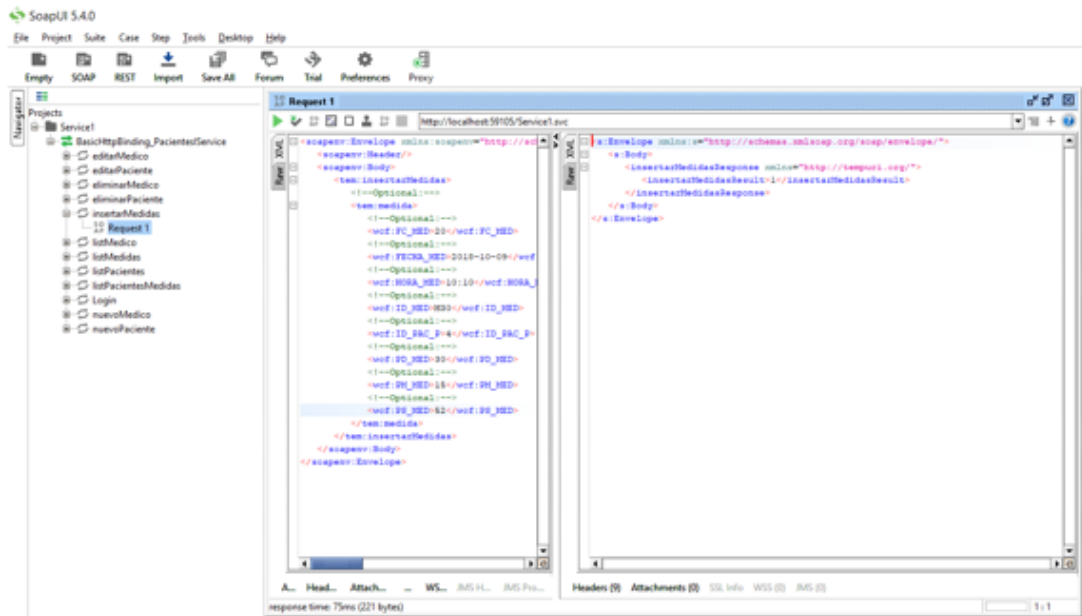


Figura 32: Test servicio web - método para insertar un dato desde la app móvil
Elaborado por: Edison Chisaguano

Creación de la aplicación de gestión de información de pacientes

Para la creación de la aplicación seleccionamos agregar nuevo sitio web vacío de ASP.NET el cuál es desarrollado para realizar los diferentes procesos de gestión de información de los pacientes, ahí se trabaja con un una plantilla que nos sirve para mantener un mismo diseño en todo el sistema.

Fragmento de código de la pagina maestra de la aplicación

```

<asp:ContentPlaceHolder id="head" runat="server">
</asp:ContentPlaceHolder>
</head>
<body>
<form id="form1" runat="server">
<nav class="navbar navbar-expand-lg navbar-dark bg-primary" role="navigation">
<div class="container">
Pacientes Hipertensos</a>
<button class="navbar-toggler border-0" type="button" data-toggle="collapse" data-target="#exCollapsingNavbar">
&#9776;
</button>
<div class="collapse navbar-collapse" id="exCollapsingNavbar">
<ul class="nav navbar-nav">
<li class="nav-item active" id="menpacientes"><a href="Mpacientes.aspx" class="nav-link"><i class="fa fa-user-m
<li class="nav-item active" id="menmedicos"><a href="Mmedicos.aspx" class="nav-link"><i class="fa fa-user-m
<li class="nav-item active" id="menumedidas"><a href="Mmedidas.aspx" class="nav-link"><i class="fa fa-bar-cl

</ul>
<ul class="nav navbar-nav flex-row justify-content-between ml-auto">
<li class="nav-item disabled"> <a id="usuario" href="javascript: void(0)" class="nav-link"><i class="fa fa
<li class="dropdown order-1">
<asp:Button ID="btnSalir" runat="server" Text="Salir" CssClass="btn btn-danger btn-sm" style="margin-top:
</div>

```

Figura 33: Fragmento de código de la página maestra de la aplicación
Elaborado por: Edison Chisaguano

Implementación de la gráfica estadística para el monitoreo de la presión arterial

Highcharts es una librería de gráficas dinámicas, multiplataforma desarrollado en lenguaje javascript, usa de igual manera jquery y ajax para poder ser flexible e interactivas con el usuario.

Highcharts permite realizar diferentes tipos de gráficas ya que admite línea, spline, área, medidores angulares, área spline, columnas, barras, circular, polar chart entre otras. Las gráficas son responsive ya que cuentan con la capacidad de respuesta para ajustar el gráfico a las dimensiones de su contenedor, pero automáticamente encuentra una ubicación óptima para elementos que no son gráficos, como leyendas y encabezados.

Identificación de cada una de las partes que comprende la gráfica estadística

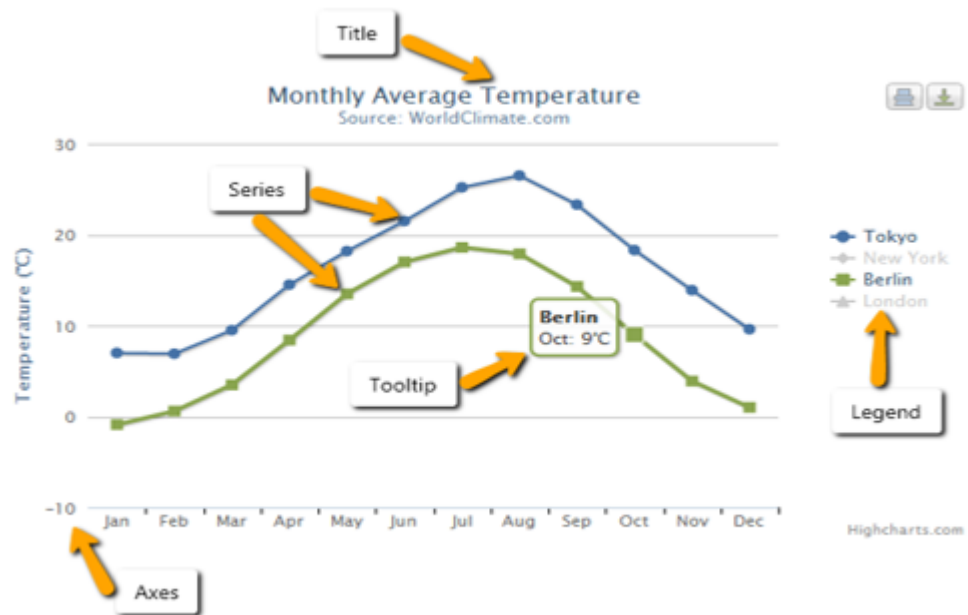


Figura 34: Identificación partes de gráfica hightcharts
Elaborado por: Edison Chisaguano

Highstock de igual forma en una librería basa en Highcharts, lo que significa que tiene toda la funcionalidad principal de Highcharts, además de contar con algunas características adicionales para una visualización más técnica y sofisticada.

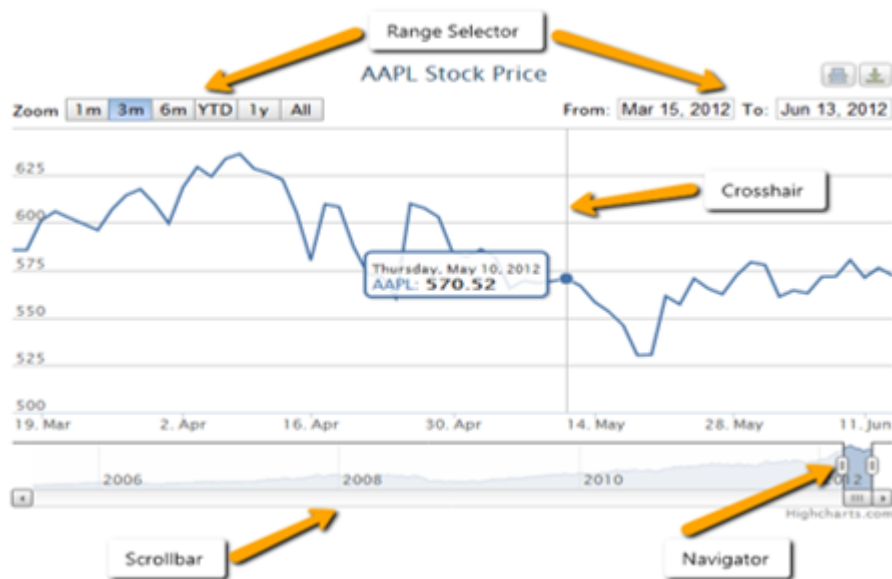


Figura 35: Identificación partes de gráfica highstock
Elaborado por: Edison Chisaguano

Navegador (Navigator): permite ajustar el rango del gráfico que se desea mostrar.

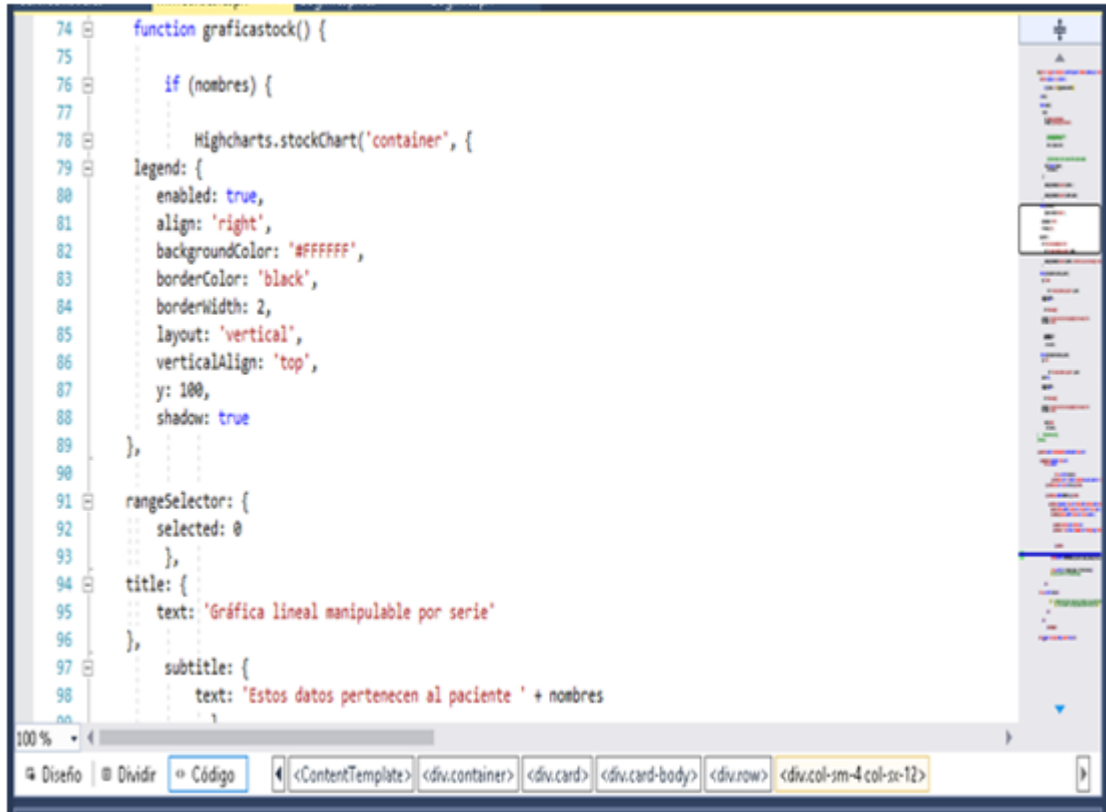
Selector de rango (Range selector): le permite al usuario seleccionar rápidamente un rango para mostrar en el gráfico o especificar el intervalo exacto que se visualizara.

Barra de desplazamiento (Scrollbar): permite desplazarse por el gráfico de manera dinámica

Cruz (Crosshair): muestra una línea siguiendo la información para leer mejor los resultados del eje x. Esta funcionalidad se puede encontrar en la opción Tooltip. Los puntos de mira también se pueden usar en Highcharts, pero no están habilitados por defecto.

En el desarrollo de la gráfica para poder identificar cada uno de los puntos con su detalla información se trabajó con la librería highstock ya que es gratuita, y permite la manipulación de su código fuente para su implementación.

Extracto de código de la implementación de la gráfica estadística



```
74 function graficastock() {
75
76     if (nombres) {
77
78         Highcharts.stockChart('container', {
79             legend: {
80                 enabled: true,
81                 align: 'right',
82                 backgroundColor: '#FFFFFF',
83                 borderColor: 'black',
84                 borderWidth: 2,
85                 layout: 'vertical',
86                 verticalAlign: 'top',
87                 y: 100,
88                 shadow: true
89             },
90
91             rangeSelector: {
92                 selected: 0
93             },
94             title: {
95                 text: 'Gráfica lineal manipulable por serie'
96             },
97             subtitle: {
98                 text: 'Estos datos pertenecen al paciente ' + nombres
99             }
100         });
101     }
102 }
```

Figura 36: Código gráfica
Elaborado por: Edison Chisaguano

4.4. Pruebas

En esta fase se realizó las pruebas correspondientes al sistema, ya que son imprescindibles en la metodología de desarrollo de software, en esta se detecta y corrige los errores en el software comprobando el correcto funcionamiento de la aplicación web.

4.4.1. Pruebas de caja blanca

Las pruebas de caja blanca en el desarrollo de software son denominadas así porque se las realiza a las funciones internas, es decir directamente al código fuente de la aplicación informática llevando a cabo un análisis completo por cada uno de los caminos posibles que sigue el mismo y así comprobar el funcionamiento de la aplicación y poder garantizar la confiabilidad de la misma.

Las pruebas de caja blanca se encuentran ligadas al código fuente, estas examinan las funciones, asegurando que durante la prueba se hayan ejecutado todas las

sentencias programadas de la aplicación por lo menos una vez. Con el método de prueba del camino básico se puede obtener la complejidad del diseño y servir como guía para que una función tome una serie de caminos básicos los cuales deben ejecutarse al menos una vez.

Código fuente del método para consultar los datos de las medidas de la presión arterial del paciente

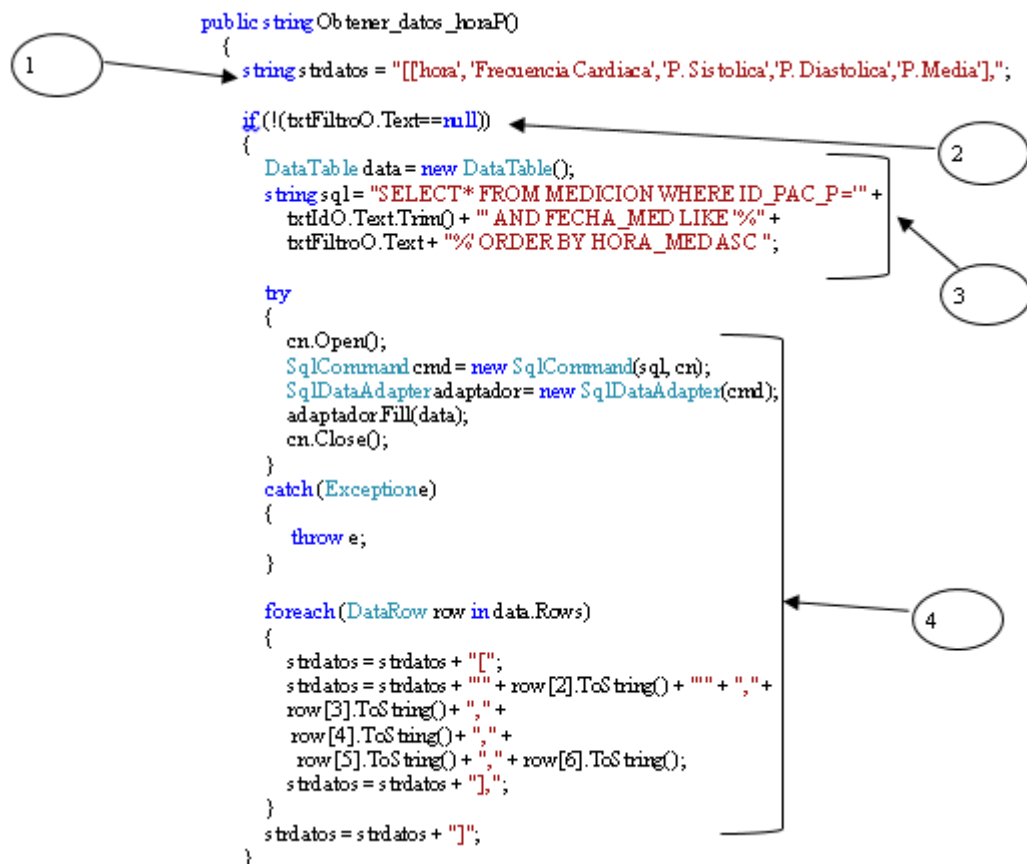


Figura 37: Camino básico del método para consultar las medidas PA (a)
Elaborado por: Edison Chisaguano

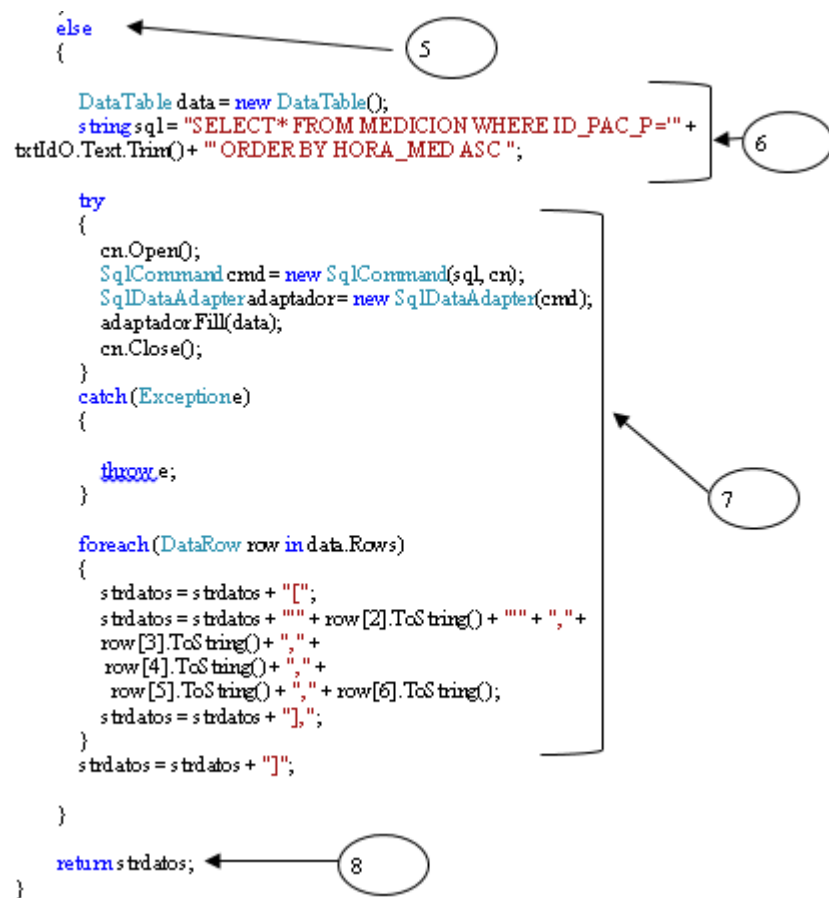


Figura 38: Camino básico del método para consultar las medidas PA (b)
 Elaborado por: Edison Chisaguano

Grafo de flujo

Mediante el grafo de flujo se observa los caminos independientes que puede tener los procesos en la ejecución/compilación del código fuente.

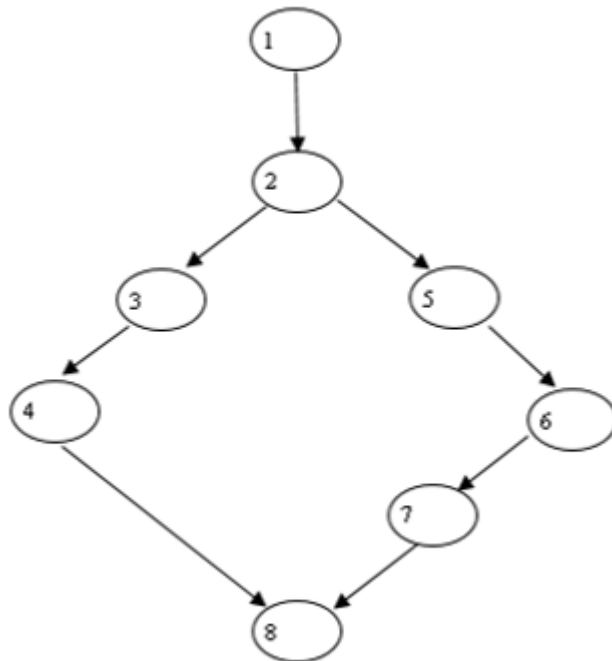


Figura 39: Grafo de flujo del método para consultar las medidas PA
Elaborado por: Edison Chisaguano

Tabla 14: Cuadro de pruebas: Método para consulta de datos de medición

Elaborado por: Edison Chisaguano

Proceso	Caminos posibles
1. Variable tipo cadena para almacenar los datos de las medidas de la PA. 2. Filtra las medidas por fecha 3. Consulta con filtro a la BDD 4. Ejecuta la consulta a la BDD. 5. No filtra las medidas 6. Consulta sin filtro a la BDD. 7. Ejecuta la consulta sin filtro a la BDD 8. Devuelve las medidas de la PA del paciente.	Camino 1: 1-2-3-4-8 Camino 2: 1-2-5-6-7-8

Complejidad ciclomática

La complejidad ciclomática es la métrica que ayuda a determinar la complejidad lógica de un programa estructurado.

Tabla 15: Complejidad ciclomática del método consulta medidas.

Elaborado por: Edison Chisaguano

Fórmula	Descripción
$V(G)=a-n+2$ $V(G)=8-8+2$ $V(G)=2 \rightarrow$ Método sencillo, poco riesgo.	a número de aristas del grafo, n número de nodos.
$V(G)=r+1$ $V(G)=1+1$ $V(G)=2$	r, representa el número de regiones cerradas del grafo.
$V(G)=c+1$ $V(G)=1+1$ $V(G)=2$	c, nodos de condición.

4.4.2. Prueba de caja negra

La prueba de caja negra se las realiza a la parte de las salidas de datos es decir al funcionamiento de la aplicación sin tomar en cuenta el código fuente del sistema. En las siguientes imágenes se detalla el funcionamiento de cada módulo a los cuales tienen acceso los usuarios.

Inicio de sesión

En login del sistema se autentifica el usuario de igual forma se realiza el proceso para identificar que rol tiene, para poder acceder a los diferentes módulos, teniendo en cuenta el nivel de accesibilidad de cada uno.



Figura 40: Inicio de sesión
Elaborado por: Edison Chisaguano

Inicio de sesión contiene controles que permite iniciar una sesión digitando el nombre de usuario y la contraseña.

Rol administrador

Página de inicio identificado como administrador tiene acceso al módulo de pacientes, médicos y medidas donde se realiza el control de la presión arterial de un paciente y también se puede generar reportes.



Figura 41: Página rol administrador
Elaborado por: Edison Chisaguano

Rol médico

Se controla el nivel de accesibilidad y tiene permiso para el módulo de medidas.



Figura 42: Página rol médico
Elaborado por: Edison Chisaguano

Módulo de pacientes

El usuario identificado como administrador puede realizar el mantenimiento de la información que será almacenada en la base de datos

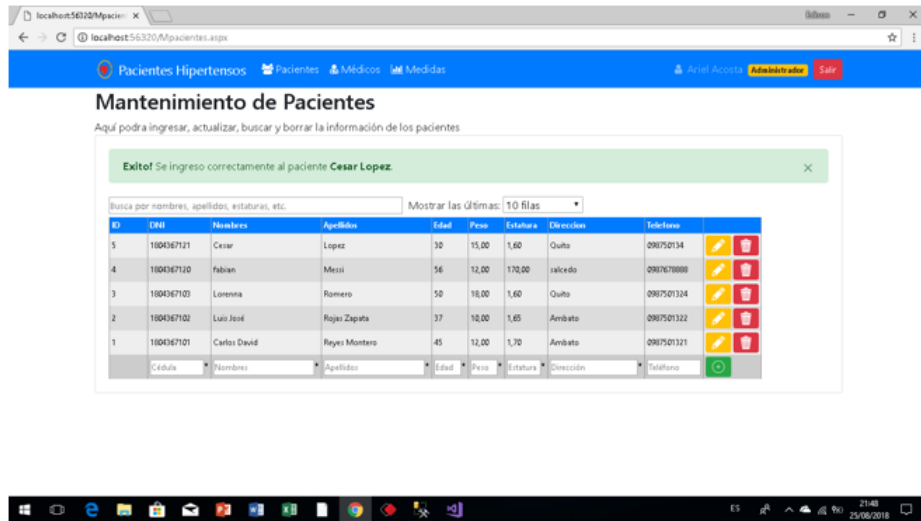


Figura 43: Página mantenimiento pacientes.
Elaborado por: Edison Chisaguano

La página de pacientes cuenta con controles de campos requeridos, emite notificaciones al realizar los procesos de insertar, actualizar y eliminar un registro de la base de datos.

Notificación emitida del sistema



Figura 44: Notificación del sistema
Elaborado por: Edison Chisaguano

Notificación al eliminar un registro

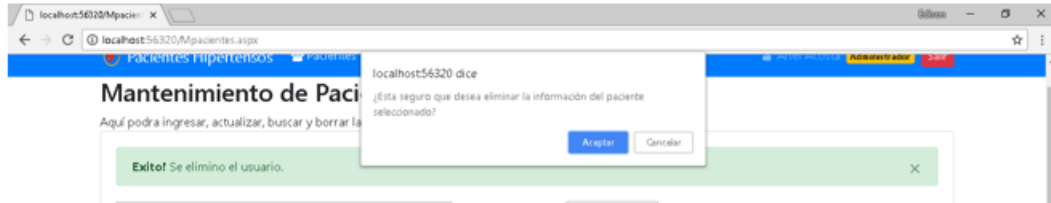


Figura 45: Notificación al eliminar un registro
Elaborado por: Edison Chisaguano

Búsquedas inteligentes

En las búsquedas inteligentes se las puede filtrar por el campo que el usuario desee reduciendo así los registros de la lista, en la siguiente imagen se filtra por el campo dirección donde reside un paciente.

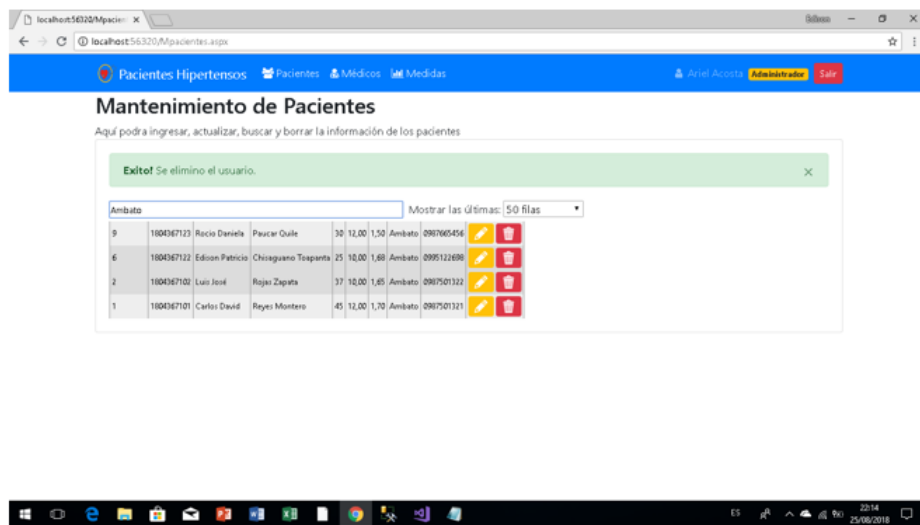


Figura 46: Búsqueda inteligente
Elaborado por: Edison Chisaguano

Al actualizar un registro se nos habilita los campos de texto de la tabla y muestra su respectivo mensaje de confirmación.

Exito! Se actualizo correctamente al paciente cuyo ID= 9. X

Ambato Mostrar las últimas: 50 filas

ID	DNI	Nombres	Apellidos	Edad	Peso	Estatura	Dirección	Teléfono		
11	1804367125	Jessica	Villa	45	15,00	1,60	Salcedo	087653535		
10	1804367124	Lenin	Correa	39	15,00	1,60	Quito	0987089900		
9	1804367123	Rocio Daniela	Paucar Quile	30	12,00	1,50	Ibarra	0987645456		
6	1804367122	Edison Patricio	Chisaguano Toapanta	25	10,00	1,68	Ambato	0995122698		
5	1804367121	Cesar	Lopez	30	15,00	1,60	Quito	098750134		
4	1804367120	Fabian	Messi	56	12,00	170,00	Salcedo	0987678888		
3	1804367103	Lorena	Romero	50	18,00	1,60	Quito	0987501324		
2	1804367102	Luis José	Rojas Zapata	37	10,00	1,65	Ambato	0987501322		
1	1804367101	Carlos David	Reyes Montero	45	12,00	1,70	Ambato	0987501321		

Cédula: Nombre: Apellidos: Edad: Peso: Estatura: Dirección: Teléfono:

Figura 47: Registro actualizado
Elaborado por: Edison Chisaguano

Módulo de medidas

Muestra la lista de pacientes y medidas de la presión arterial las cuales se encuentra almacenados en la base de datos de igual manera cuenta con buscador inteligente que ayuda agilizando el proceso de busque de un registro, en la lista de pacientes se introdujo un botón con el cual se filtra solo las medidas de un paciente en específico para luego pasarlas una gráfica estadística.

Pacientes Hipertensos Pacientes Médicos Medicinas Ariel Acosta Administrador Salir

Pacientes
Seleccione el paciente a monitorear

Busca por dni, nombres Mostrar las últimas: 10 filas

ID	DNI	NOMBRES	
11	1804367125	Jessica Villa	<input type="checkbox"/>
10	1804367124	Lenin Correa	<input type="checkbox"/>
9	1804367123	Rocio Daniela Paucar Quile	<input checked="" type="checkbox"/>
6	1804367122	Edison Patricio Chisaguano Toapanta	<input checked="" type="checkbox"/>
5	1804367121	Cesar Lopez	<input checked="" type="checkbox"/>
4	1804367120	Fabian Messi	<input checked="" type="checkbox"/>
3	1804367103	Lorena Romero	<input checked="" type="checkbox"/>
2	1804367102	Luis José Rojas Zapata	<input checked="" type="checkbox"/>
1	1804367101	Carlos David Reyes	<input checked="" type="checkbox"/>

Medidas
Tabla de medidas del paciente seleccionado, ordenado por fecha y hora

CODIGO	FECHA	HORA	FC	PS	PD	PM
M04	05 mar. 2018	10:10:12	60	130	90	100
M05	05 mar. 2018	11:00:10	75	150	120	120
M06	06 mar. 2018	07:00:34	50	110	70	70
M07	06 mar. 2018	08:00:23	90	120	150	85
M08	06 mar. 2018	08:30:12	55	80	120	70
M09	06 mar. 2018	09:10:34	75	135	130	90
	10 mar.					

FC: Frecuencia Cardíaca PS: Presión Sistólica
PD: Presión Diastólica PM: Presión Media

Figura 48: Página de medidas
Elaborado por: Edison Chisaguano

Gráfica estadística para el monitoreo de la presión arterial

En la gráfica estadística se puede identificar cual es el valor de la frecuencia cardíaca, presión sistólica, presión diastólica y presión media de un paciente de manera visual, de tal forma que cada punto se encuentra distribuido y clasificado

de acuerdo a la fecha y hora que fue emitida la medición gracias a la librería utilizada también se puede apreciar el día en la que se realizó el paciente.

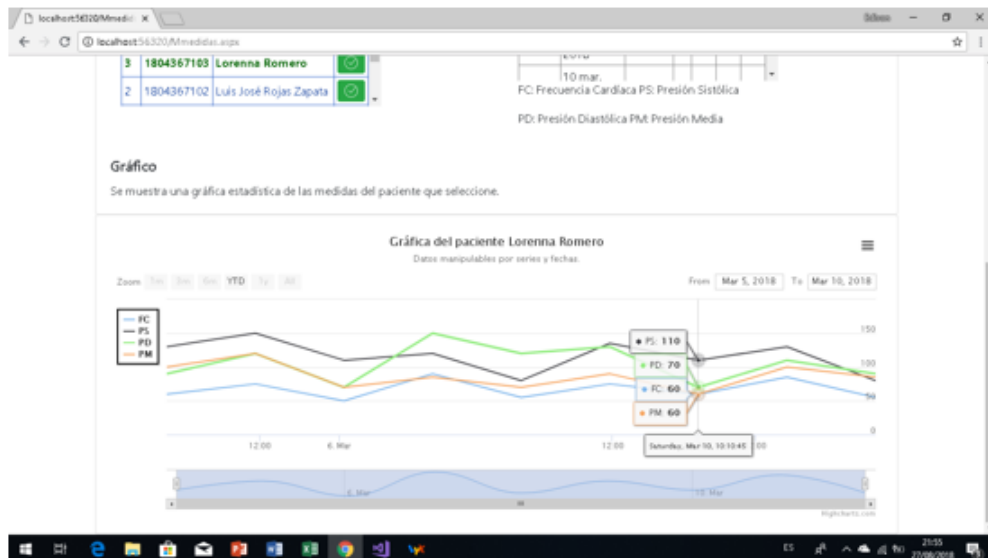


Figura 49: Gráfica estadística de la presión arterial
Elaborado por: Edison Chisaguano

Manipulación de las series

La gráfica se la puede manipular a conveniencia del usuario con el fin de observar detenida y detalladamente el comportamiento de una medida toma del paciente ayudando así en el análisis de presión arterial, para seleccionar que serie mostrar el usuario debe dar click sobre las leyendas de la gráfica identificadas por colores al lado derecho de la aplicación.

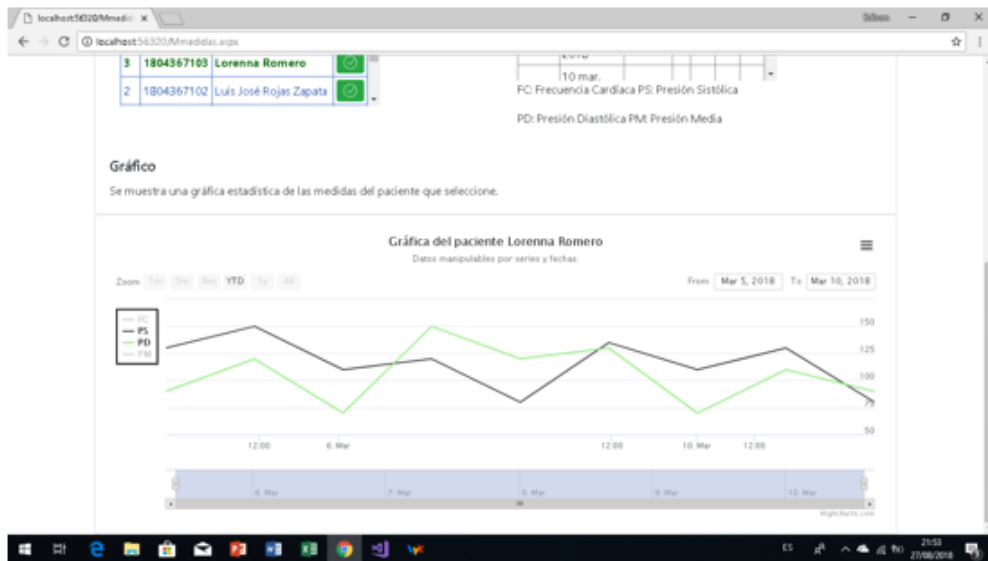


Figura 50: Manipulación de series
Elaborado por: Edison Chisaguano

La gráfica cuenta en la parte inferior con una barra desplazable la cual permite filtrar por fecha la medida de un paciente.

Filtración de medidas por fecha

El usuario puede seleccionar un rango de fechas específico para el análisis de la presión arterial de un paciente.

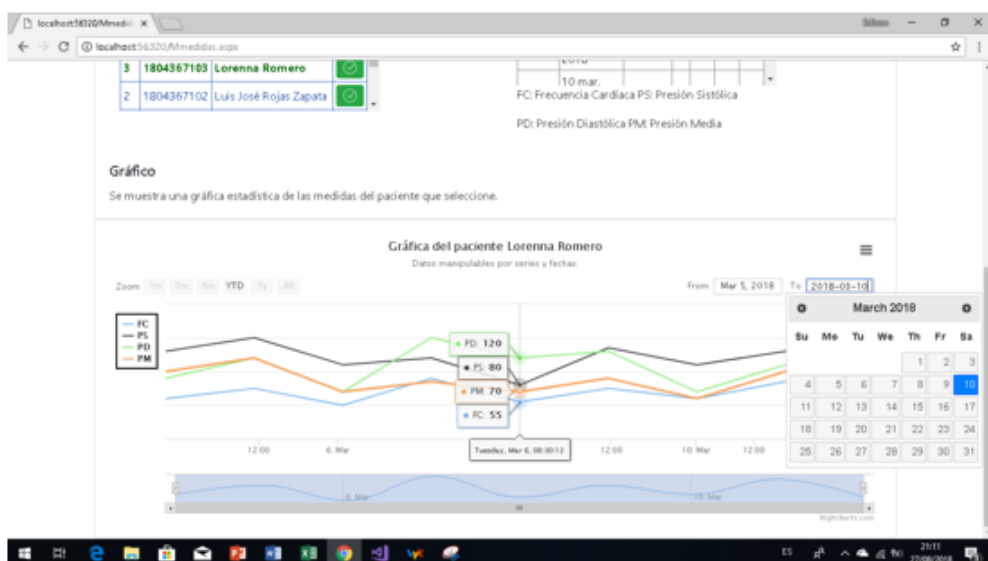


Figura 51: Filtrado de fechas
Elaborado por: Edison Chisaguano

Opciones

Con las opciones de exportación, los usuarios pueden exportar el cuadro a formato PNG, JPG, PDF o SVG con solo hacer clic en un botón, o imprimir el cuadro directamente desde la página web

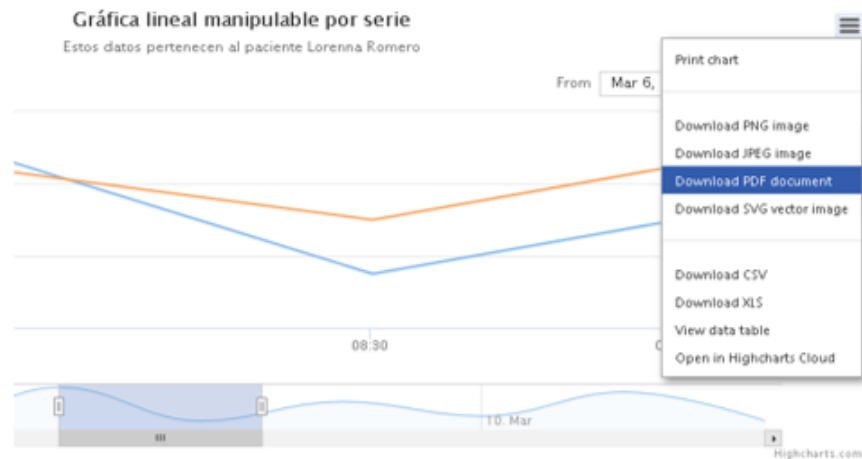


Figura 52: Opciones de exportación
Elaborado por: Edison Chisaguano

Reporte

Reporte en forma gráfica del comportamiento de las medidas tomadas de un paciente.

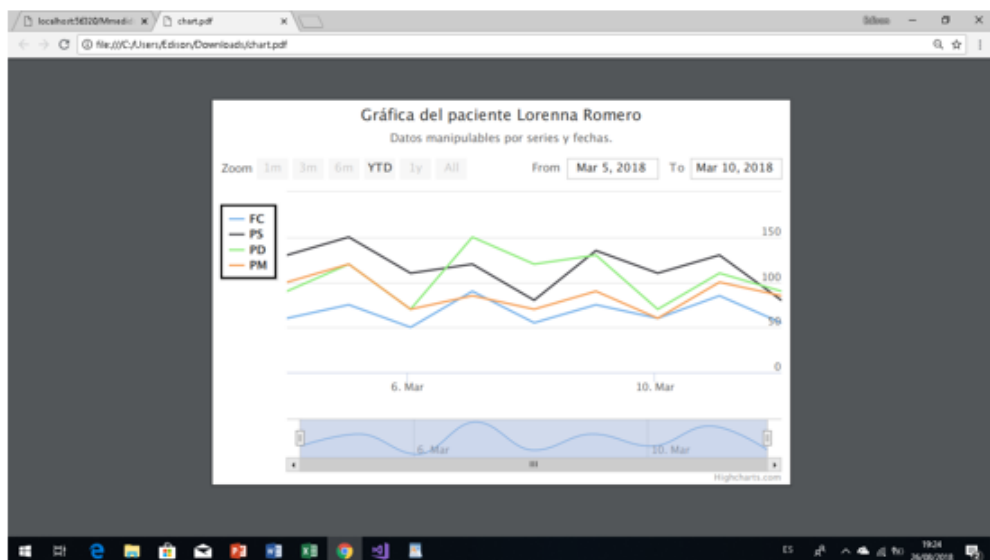


Figura 53: Reporte estadístico gráfico de monitoreo presión arterial
Elaborado por: Edison Chisaguano

4.4.3. Pruebas de rendimiento

Se realizó pruebas de rendimiento para poder visualizar la capacidad de tiempo de respuesta que tiene la aplicación de monitoreo de pacientes para dicha prueba se utilizó la aplicación jmeter.

En la prueba de rendimiento con la ayuda de la aplicación jmeter se simuló que 50 usuarios ingresan al sistema en 5 segundos obteniendo como resultado la siguiente tabla.

Tabla 16: Cuadro de pruebas: Resultados - rendimiento

Elaborado por: Edison Chisaguano

Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Latency	Sent Bytes	Conned Time(ms)	Status
1	18:40:28.759	Usuarios 1-1	PeticonPruebas	5	5	113	2	
2	18:40:29.258	Usuarios 1-2	PeticonPruebas	9	9	113	3	
3	18:40:29.761	Usuarios 1-3	PeticonPruebas	10	10	113	3	
4	18:40:30.260	Usuarios 1-4	PeticonPruebas	7	7	113	2	
5	18:40:30.763	Usuarios 1-5	PeticonPruebas	8	8	113	4	
6	18:40:31.269	Usuarios 1-6	PeticonPruebas	10	10	113	3	
7	18:40:31.765	Usuarios 1-7	PeticonPruebas	4	4	113	1	
8	18:40:32.265	Usuarios 1-8	PeticonPruebas	6	6	113	2	
9	18:40:32.765	Usuarios 1-9	PeticonPruebas	6	6	113	2	
10	18:40:33.266	Usuarios 1-10	PeticonPruebas	6	6	113	2	

Tabla 17: Cuadro de pruebas: Resultados - rendimiento (a)

Elaborado por: Edison Chisaguano

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Avg Bytes
PeticonPruebas	50	26	1	454	87.92	0.00%	10.2/sec	943.0
TOTAL	50	26	1	454	87.92	0.00%	10.2/sec	943.0

Como puede verse, el tiempo promedio para acceder a la página es de 26 milisegundos, realizándose un total de 50 requerimientos al servidor, además su rendimiento es de 10.2 segundos demostrando así que la velocidad respuesta al realizar un proceso la app es favorable para trabajar sin problema alguno.

El sistema web responde de manera correcta a todas las pruebas efectuadas, garantizando un software funcional y útil para la gestión de información y monitoreo de la presión arterial de un paciente.

CAPÍTULO 5

Conclusiones y Recomendaciones

5.1. Conclusiones

- Con la aplicación informática se pretende que el usuario almacene datos sobre la hipertensión arterial con el fin de facilitar el control y seguimiento del estado de salud arterial mostrando así una gráfica donde se puede visualizar el comportamiento de las medidas de un paciente en un determinado tiempo.
- Al llevar a cabo el presente trabajo de gestión de información se logró sistematizar la lectura de la frecuencia cardíaca, presión sistólica, diastólica y presión media de un paciente de forma muy ágil, ayudando así en el diseño del proceso que permite identificar el grado de hipertensión arterial, mediante un software, detallando de manera eficiente un diagnóstico preventivo.
- El sistema desarrollado permite evaluar el resultado de la lectura cardíaca obtenida, mediante una tabla informativa que contiene las mediciones tomadas, mostrándolas en el sistema, donde le indicará el rango que se encuentra el estado de salud de un paciente.
- Se ha logrado cumplir con los requisitos funcionales y no funcionales alcanzando así que la aplicación sea de fácil uso, intuitiva, segura y rápida.
- Al utilizar la arquitectura orientada a servicios (SOA) los recursos en una red están disponibles como servicios independientes a los que se puede acceder sin tener conocimiento de la implementación de su plataforma permitiendo así la integración de aplicaciones y coordinar los procesos de una manera efectiva y rápida.

5.2. Recomendaciones

- Se sugiere la investigación en cuanto a la implementación de un proceso que le informe cuando una persona tiene hipertensión y notifique que plan

llevar para el control de la presión arterial, y así mejorar la calidad de vida del paciente.

- Como trabajos de análisis a posterior se recomienda el estudio de librerías pagadas y gratuitas para gráficas técnicas de estadística para poder implementarlas en aplicaciones web relacionadas a la ciencia de la salud.
- Se recomienda efectuar respaldos (copias) de información periódicamente de la Base de Datos y almacenarlos en medios físicos externos para salvaguardar la integridad de la información almacenada en el sistema.

Bibliografía

- [1] L. J. P. Arredondo, “Las nuevas tecnologías de la información (ntic) en la medicina: la telemedicina en cuba,” *Revista Cubana de Informática Médica*, 2006.
- [2] H. V. Roldán González, “Desarrollo de un modelo de gestión de servicios de ti para aplicaciones de telemedicina en el ecuador,” Master’s thesis, Quito, 2016., 2016.
- [3] L. E. M. Ramírez, “Evaluación de exámenes químicos y hematológicos en pacientes con hipertensión arterial del centro de salud n°1 del cantón ambato,” 2012.
- [4] M. Nielsen and V. Villarreal, “Diseño y desarrollo de una aplicación móvil que facilite el seguimiento y control de pacientes hipertensos en panamá,” in *Memorias de Congresos UTP*, pp. 54–62, 2017.
- [5] B. G. Criollo Altamirano, *Desarrollo de un Sistema de Monitoreo y Medición del Ritmo Cardíaco para personas hipertensas a través de Dispositivos Móviles con Sistema Operativo Android*. PhD thesis, Universidad de Guayaquil. Facultad de Ciencias Matemáticas y Físicas. Carrera de Ingeniería en Sistemas Computacionales. Carrera de Ingeniería en Networking y Telecomunicaciones., 2014.
- [6] C. Cadavid Gómez, “Qué es la hipertensión. 24,” 1990.
- [7] A. Delgado, L. González, and F. Piedrabuena, “Desarrollo de aplicaciones con enfoque soa (service oriented architecture),” *Reportes Técnicos 06-16*, 2006.
- [8] M. C. Franky, “Entendiendo el desarrollo de los sistemas soa,” *Editorial Revista Sistemas. ACIS*, no. 111, 2009.
- [9] P. Bazán, “Bpel: una propuesta para el uso de web services,” in *XIII Congreso Argentino de Ciencias de la Computación*, 2007.
- [10] U. de Murcia, “Metodologías de desarrollo de software,” 2006. [Online] Available: <http://www.um.es/docencia/barzana/IAGP/Iagp2.html>.

- [11] M. Trigás Gallego, “Metodología scrum,” 2012.
- [12] C. Ble, “Modelo en cascada,” 2013.
- [13] E. Parra Castrillón, “Propuesta de metodología de desarrollo de software para objetos virtuales de aprendizaje-mesova,” *Revista Virtual Universidad Católica del Norte*, no. 34, 2011.
- [14] J. Joskowicz, “Reglas y prácticas en extreme programming,” *Universidad de Vigo*, vol. 22, 2008.
- [15] I. Grupo, “Metodologías ágiles en el desarrollo de software,” *VIII Jornadas de Ingeniería del Software y Bases de Datos*, 2003.
- [16] N. I. Scagnoli, “El aula virtual: usos y elementos que la componen,” 2000.
- [17] K. Kendall and J. Kendall, “Análisis y diseño de sistemas. (octava edición),” 2011.
- [18] F. Berzal, J. C. Cubero, and F. J. Cortijo, *Desarrollo profesional de aplicaciones web con Asp. net*. iKor Consulting, 2005.
- [19] Óscar Garrido Chueco, “Diseño del frontend del panel de control de un ayuntamiento,” 2017.
- [20] J. Hugon, *C# 6: Desarrolle aplicaciones Windows con Visual Studio 2015*. Ediciones ENI, 2015.
- [21] O. Valle and O. Rivera, “Monitoreo e indicadores,” *IDIE Guatemala. Educación Inicial y Derechos de la Infancia*. Sevilla: Junta de Andalucía, 2008.
- [22] C. O. Ruiz, “El sistema cardiovascular,” 2012. [Online] Available: https://webs.um.es/clara/miwiki/doku.php?id=sistema_cardiovascular.
- [23] A.D.A.M., “Ataque cardíaco,” 2015. [Online] Available: <https://medlineplus.gov/spanish/ency/article/000195.htm>.
- [24] J. Turrent, L. Talledo, A. González, J. Gundián, and C. Remuñán, “Comportamiento y manejo de la enfermedad cerebrovascular en una unidad de cuidados intensivos,” *Revista Cubana de Medicina Intensiva y Emergencias (periódica en línea)*, vol. 3, no. 2, 2004.

- [25] A. M. P. y Cristina Diéguez Varela, “Hipertension arterial,” 2016. [Online] Available: <http://cardiosaudeferrol.com/hipertension-arterial/>.
- [26] C. S. y M.T. Aguilera, “Evaluación clínica del paciente hipertenso,” 2001. [Online] Available: <http://docplayer.es/66746351-Evaluacion-clinica-del-paciente-hipertenso.html>.
- [27] M. Canto, D. Pereda, and A. Seguro, “Service oriented architecture (soa),” *Facultad de Ingenieria, Universidad de la República, Montevideo, Uruguay*, 2006.
- [28] F. Berzal, J. C. Cubero, and F. J. Cortijo, *Desarrollo profesional de aplicaciones web con Asp. net*. iKor Consulting, 2005.
- [29] M. F. y Michael R Gillings, “Una comparación de lenguajes de programación usados en bioinformática,” 2013. [Online] Available: <https://goo.gl/fh7Krq>.

Anexos y Apéndices

Anexo A

Anexo A

Manual de usuario

El sistema que permitirá el monitoreo de pacientes, a través de una gráfica estadística tiene como finalidad detallar de manera eficiente un diagnóstico sobre la hipertensión arterial, guardando datos relevantes sobre su salud y así facilitar el seguimiento y control de su enfermedad.

Sistema web

Ingreso al sistema



Figura 54: Manual usuario - Ingreso al sistema
Elaborado por: Edison Chisaguano

1. Etiqueta con el nombre del sistema
2. Campo para validar el usuario
3. Campo para validar la contraseña del usuario
4. Botón verde para acceder al sistema después de la respectiva validación

Pantalla de inicio

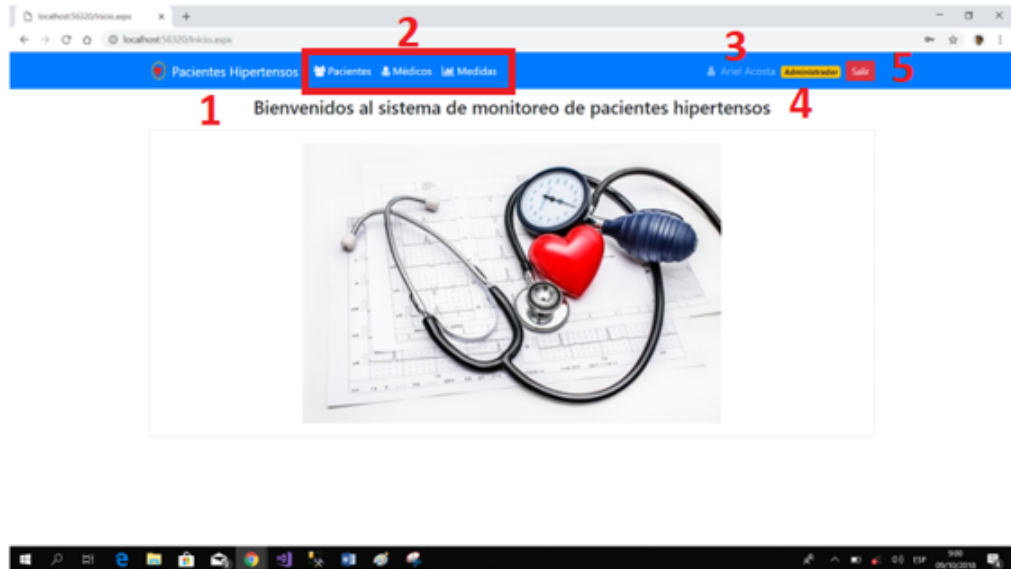


Figura 55: Manual usuario - pantalla inicio
Elaborado por: Edison Chisaguano

1. Logo del sistema – con hipervínculo a la pagina de inicio de la aplicación.
2. Menú – contiene los diferentes módulos del sistema:
 - a) Mantenimiento de pacientes
 - b) Mantenimiento de médicos
 - c) Medidas de un paciente – despliegue una lista de pacientes con sus respectivos datos (FC, PAS, PAD, PM) a monitorear.
3. Identificador de usuario que accede al sistema
4. Identificador del rol que tiene el usuario que accedió al sistema
5. Botón color rojo para salir del sistema

Mantenimiento de pacientes

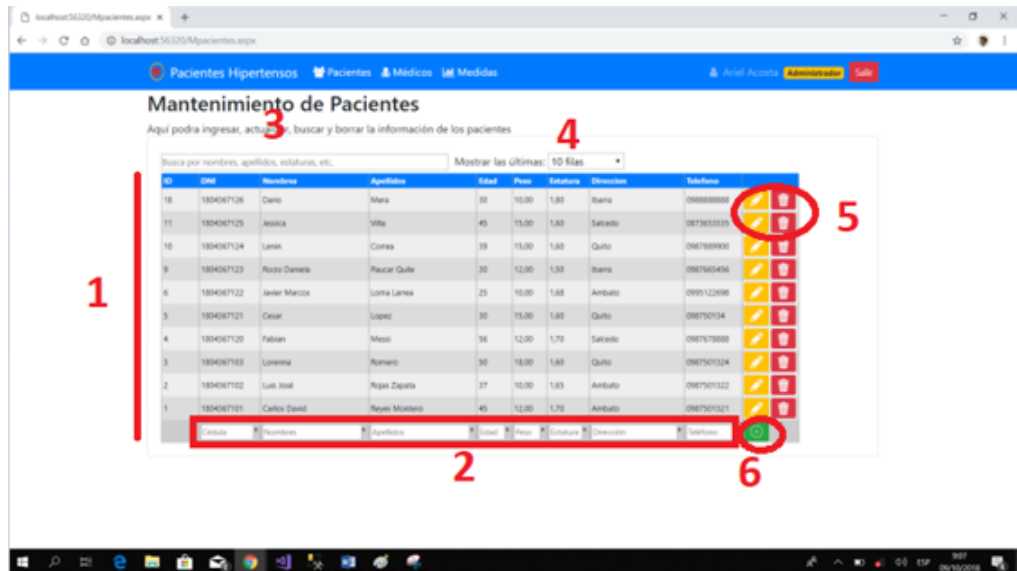


Figura 56: Manual usuario - mantenimiento pacientes
Elaborado por: Edison Chisaguano

Una vez seleccionado en el menú la opción paciente muestra lo siguiente:

1. Tabla con información (Id, cédula, nombres, apellidos, edad, peso, estatura, dirección teléfono) de pacientes ya almacenados en la base de datos
2. Campos para ingresar un registro nuevo al sistema
3. Campo para la búsqueda rápida de un registro en este caso de un paciente en particular
4. Permite seleccionar el número de registro a visualizar en la tabla
5. Botones
 - a) Color rojo para eliminar un registro
 - b) Color amarillo permite actualizar un registro de la tabla
6. Botón color verde para ingresar un nuevo registro al sistema

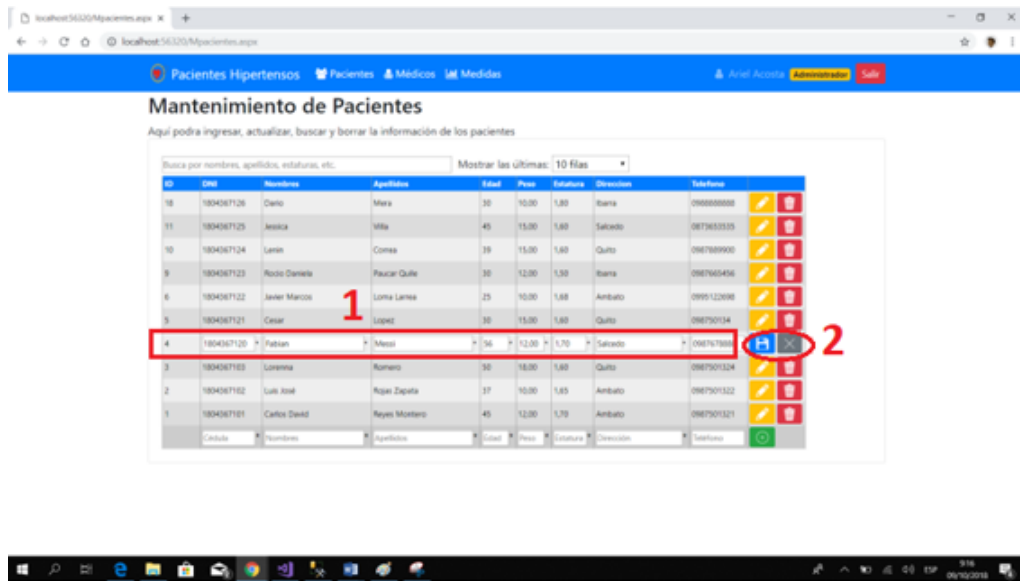


Figura 57: Manual usuario - mantenimiento pacientes (a)
Elaborado por: Edison Chisaguano

Después de haber seleccionado el botón amarillo para actualizar un registro:

1. Los campos se habilitan para modificarlo
2. Botones:
 - a) Azul para guardar el cambio del registro
 - b) Gris para cancelar la actualización del registro

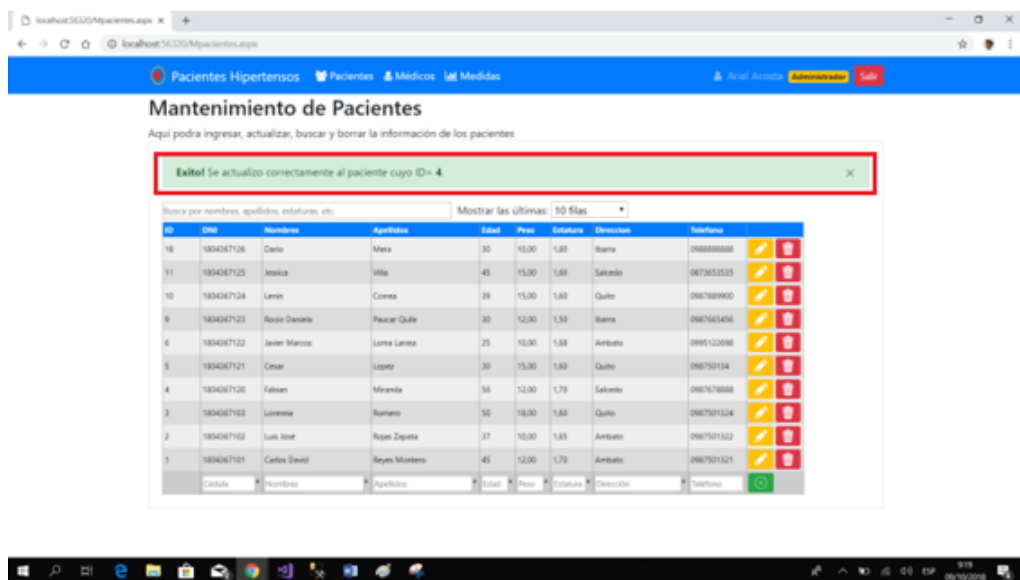


Figura 58: Manual usuario - mantenimiento pacientes (b)
Elaborado por: Edison Chisaguano

Notificación después de haber modificado un registro en el sistema

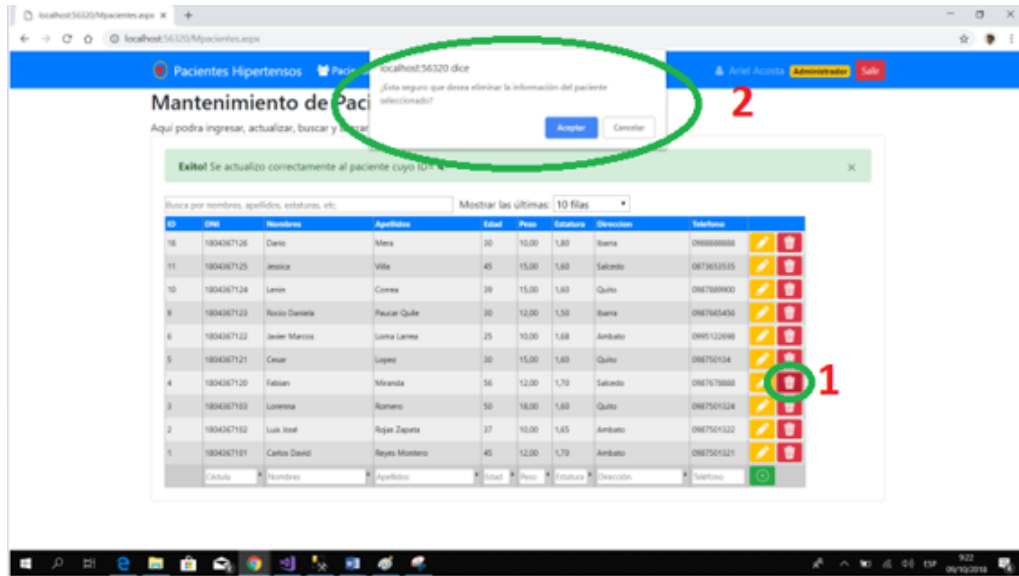


Figura 59: Manual usuario - mantenimiento pacientes (c)
Elaborado por: Edison Chisaguano

1. Botón color rojo par eliminar un registro
2. Mensaje de confirmación para eliminar el registro del sistema

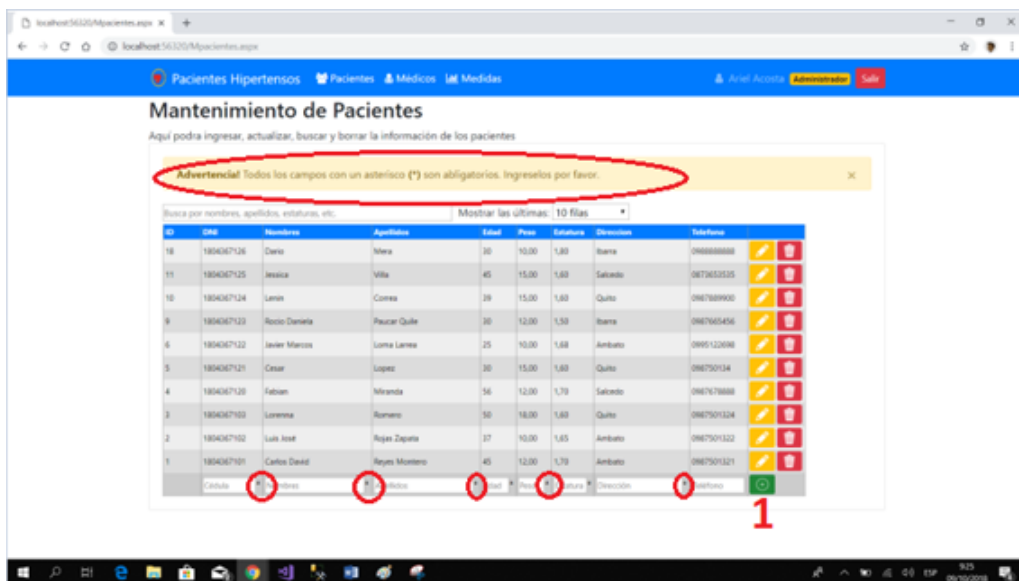


Figura 60: Manual usuario - mantenimiento pacientes (d)
Elaborado por: Edison Chisaguano

1. Botón color verde para insertar un nuevo registro, si los campos que tiene

un identificador (*) están vacíos muestra una la respectiva notificación de advertencia por lo cual debe ser ingresado al sistema.

Módulo de medidas

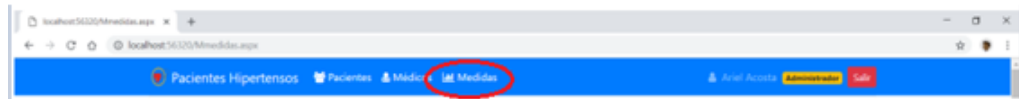


Figura 61: Manual usuario - módulo medidas
Elaborado por: Edison Chisaguano

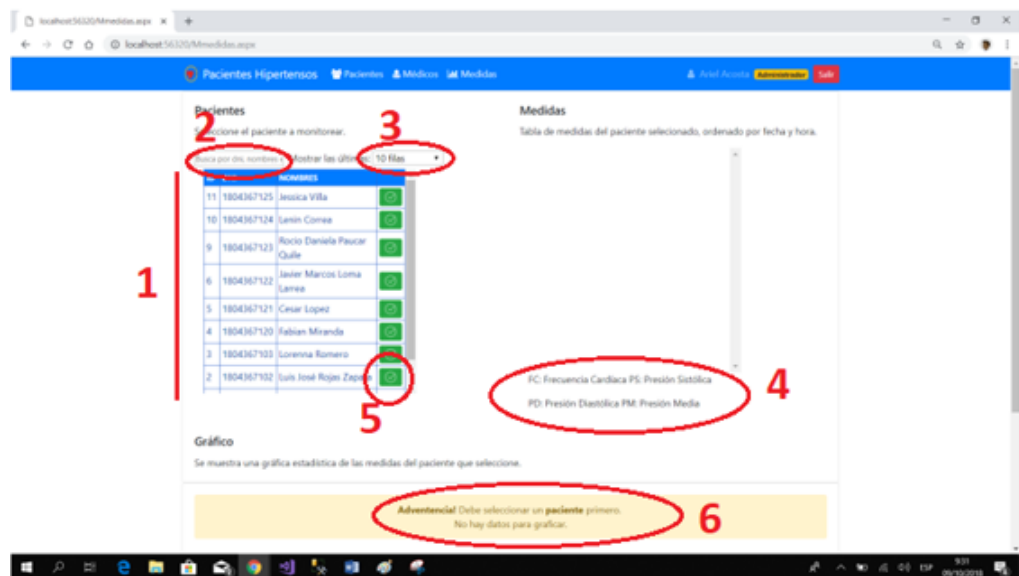


Figura 62: Manual usuario - módulo medidas (a)
Elaborado por: Edison Chisaguano

1. Tabla con la información (id, cédula y nombres) de los pacientes a monitorear
2. Campo de búsqueda inteligente para un paciente en particular
3. Se da click para seleccionar un numero de registros a visualizar
4. Etiquetas para facilitar la interpretación de los datos de la tabla medidas
5. Botón color verde para seleccionar las medidas de un paciente a ser monitoreado
6. Notificación de advertencia para que el usuario seleccione un paciente

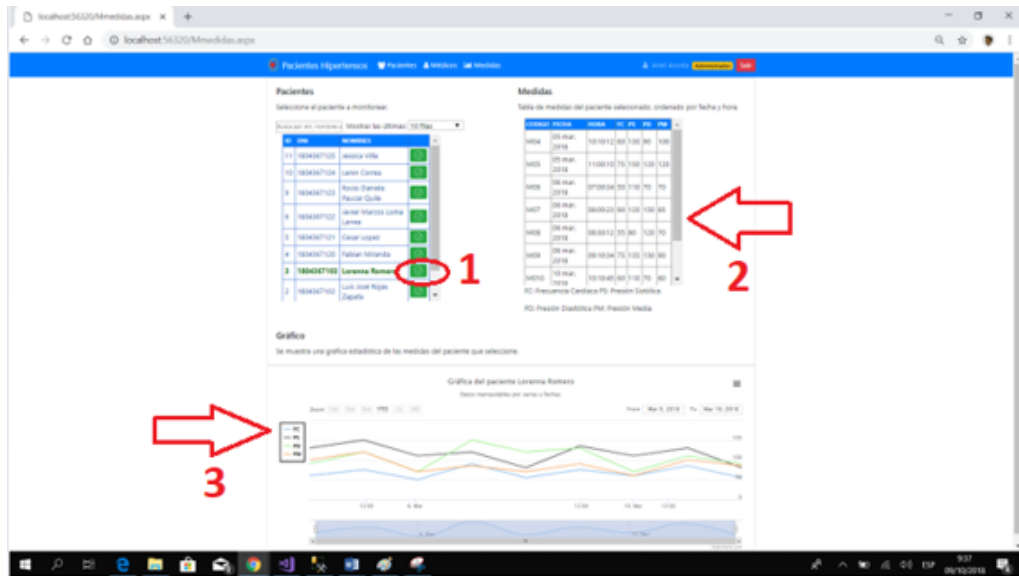


Figura 63: Manual usuario - módulo medidas (b)
Elaborado por: Edison Chisaguano

1. Botón color verde- selecciona el paciente
2. Tabla de medidas de un paciente seleccionado donde muestra la información (código, fecha que se realizó el control, la hora, FC, PS, PD, PM) detallada de todas medidas emitidas por la aplicación móvil de un paciente a monitorear la hipertensión
3. Representación gráfica de los datos obtenidos de un paciente

Representación gráfica estadística de datos de hipertensión de un paciente

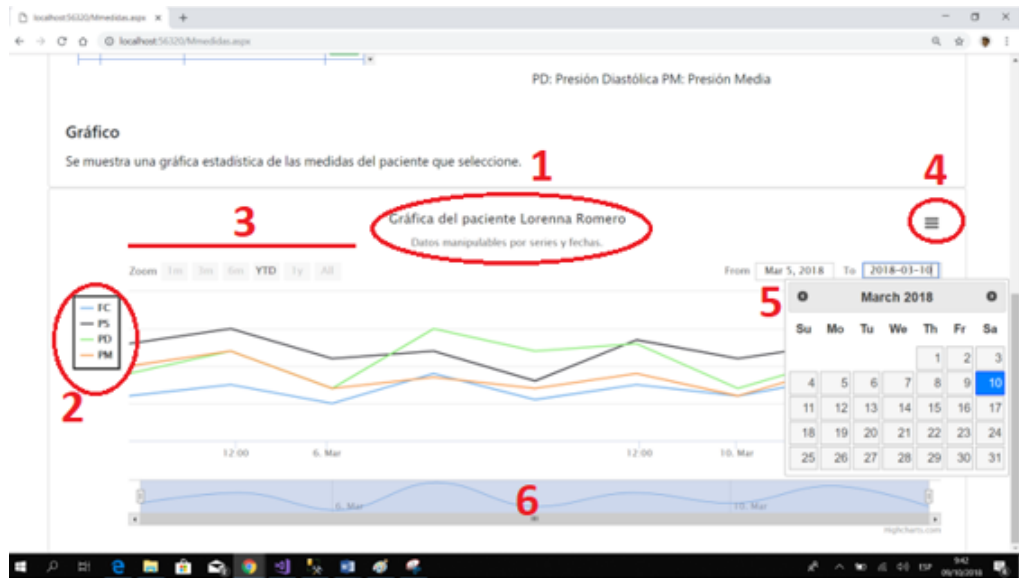


Figura 64: Manual usuario - módulo medidas (c)
Elaborado por: Edison Chisaguano

1. Identificador – nombre del paciente a monitorear
2. Leyendas – identificadas por colores de cada uno de los datos, además al dar click sobre una leyenda permite seleccionar que serie se desea observar detenidamente desactivándola o habilitándola
3. Botones que permiten ajustar el zoom de la imagen
4. Despliega una lista de opciones
5. Campos para filtrar por fecha las medidas de un paciente, al dar click sobre el campo muestra un calendario del cual se debe seleccionar una fecha en específico
6. Barra deslizable permite acercar y alejar partes de los datos, así como la panorámica a través del conjunto de datos.

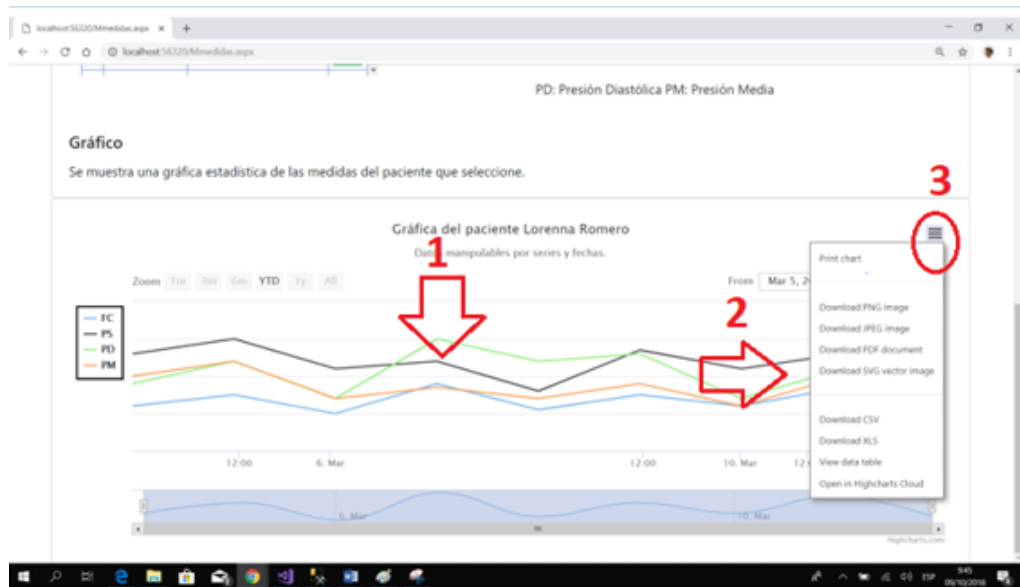


Figura 65: Manual usuario - módulo medidas (d)
Elaborado por: Edison Chisaguano

1. Representación de los datos de los pacientes en series identificadas por colores muestra cada uno de los puntos a ser analizados, emitidos por la app móvil
2. Al selecciona las opciones se despliega una lista con la cual nos permite descargar la información requerida en diferentes formatos

Anexo B

Anexo B

Certificación emitida por el Ing. Juan Pablo Pallo Mg. coordinador del proyecto de investigación.



UNIVERSIDAD TÉCNICA DE AMBATO
FACULTAD DE INGENIERÍA, ELECTRÓNICA E INDUSTRIAL
CARRERA DE INGENIERÍA EN ELECTRÓNICA Y COMUNICACIONES

CARTA DE ACEPTACIÓN

Ambato, 28 de agosto 2018

Ingeniera M.Sc.
Pilar Urrutia U.
DECANA
Facultad de Ingeniería en Sistemas, Electrónica e Industrial
Presente.

Señora Decana:

En mi calidad de coordinador del Proyecto de Investigación, por medio de la presente manifiesto a usted, que el Sr. Edison Patricio Chisaguano Toapanta, portador de la cédula de ciudadanía N° 1804367108, estudiante de Décimo nivel en el periodo académico Septiembre 2017 – Febrero 2018 de la Carrera de Ingeniería en Sistemas Computacionales e Informáticos de la Facultad Ingeniería en Sistemas, Electrónica e Industrial, tiene mi aceptación sobre el proyecto de investigación titulado "APLICACIÓN BASADA EN ARQUITECTURA ORIENTADA A SERVICIOS PARA LA GESTIÓN DE INFORMACIÓN DE PACIENTES HIPERTENSOS", el cual está vinculado al proyecto de investigación denominado "Sistema Electrónico De Detección Personas Hipertensas Mediante El Monitoreo Continuo De Bioseñales Y Alertas Preventivas Con Soporte CLOUD".

Con estos antecedentes informo que el proyecto de investigación mencionado anteriormente cumple con los objetivos establecidos, por lo expuesto anteriormente solicito se apruebe y se proceda con el trámite correspondiente.

Atentamente,



Ing. Juan Pablo Pallo, Mg.
COORDINADOR DEL PROYECTO
DE INVESTIGACIÓN

Figura 66: Certificado - aprobación del proyecto
Elaborado por: Edison Chisaguano