



**UNIVERSIDAD NACIONAL DE CHIMBORAZO
FACULTAD DE INGENIERIA
CARRERA DE TECNOLOGIAS DE LA INFORMACION**

**APLICACIÓN WEB Y MÓVIL DE VIGILANCIA PARA EL
CONSUMO DE MEDICAMENTOS EN PACIENTES
AMBULATORIOS, UTILIZANDO “FOG COMPUTING”.**

**Trabajo de Titulación para optar al título de Ingeniero en
Tecnologías de la Información**

Autor:

**Shiguango Garcia, Jonathan Cristoffer
Muñoz Sarmiento, Anderson Joel**

Tutor:

Mg. Jorge Edwin Delgado Altamirano

Riobamba, Ecuador. 2023

DECLARATORIA DE AUTORÍA

Yo, Shiguango Garcia Jonathan Cristoffer y Muñoz Sarmiento Anderson Joel, con cédulas de ciudadanía 1500782162 y 1726009101, autores del trabajo de investigación titulado: Aplicación Web y Móvil de vigilancia para el consumo de medicamentos en pacientes ambulatorios, utilizando “fog computing”, certifico que la producción, ideas, opiniones, criterios, contenidos y conclusiones expuestas son de mí exclusiva responsabilidad.

Asimismo, cedo a la Universidad Nacional de Chimborazo, en forma no exclusiva, los derechos para su uso, comunicación pública, distribución, divulgación y/o reproducción total o parcial, por medio físico o digital; en esta cesión se entiende que el cesionario no podrá obtener beneficios económicos. La posible reclamación de terceros respecto de los derechos de autor (a) de la obra referida, será de mi entera responsabilidad; librando a la Universidad Nacional de Chimborazo de posibles obligaciones.

En Riobamba, 28 de julio de 2023.



Jonathan Cristoffer Shiguango Garcia

C.I:1500782162



Anderson Joel Muñoz Sarmiento

C.I: 1726009101

DICTAMEN FAVORABLE DEL PROFESOR TUTOR



Dirección
Académica
VICERRECTORADO ACADÉMICO



ACTA FAVORABLE - INFORME FINAL DEL TRABAJO DE INVESTIGACIÓN

En la Ciudad de Riobamba, a los veinte y cuatro días del mes de julio del 2023 luego de haber revisado el Informe Final del Trabajo de Investigación presentado por los estudiantes **Jonathan Cristoffer Shiguango Garcia** con CC: **1500782162** y **Anderson Joel Muñoz Sarmiento** con CC: **1726009101**, de la carrera **Tecnologías de la Información** y dando cumplimiento a los criterios metodológicos exigidos, se emite el **ACTA FAVORABLE DEL INFORME FINAL DEL TRABAJO DE INVESTIGACIÓN** titulado **"APLICACIÓN WEB Y MÓVIL DE VIGILANCIA PARA EL CONSUMO DE MEDICAMENTOS EN PACIENTES AMBULATORIOS, UTILIZANDO "FOG COMPUTING"**, por lo tanto se autoriza la presentación del mismo para los trámites pertinentes.



Mgs. Jorge Delgado
TUTOR(A)

CERTIFICADO DE LOS MIEMBROS DEL TRIBUNAL

CERTIFICADO DE LOS MIEMBROS DEL TRIBUNAL

Quienes suscribimos, catedráticos designados Miembros del Tribunal de Grado para la evaluación del trabajo de investigación APLICACIÓN WEB Y MÓVIL DE VIGILANCIA PARA EL CONSUMO DE MEDICAMENTOS EN PACIENTES AMBULATORIOS, UTILIZANDO "FOG COMPUTING, presentado por Jonathan Cristoffer Shiguango Garcia , con cédula de identidad número 1500782162 y Anderson Joel Muñoz Sarmiento, con cedula de identidad número 1726009101 bajo la tutoría de Mgs. Jorge Edwin Delgado Altamirano; certificamos que recomendamos la APROBACIÓN de este con fines de titulación. Previamente se ha evaluado el trabajo de investigación y escuchada la sustentación por parte de su autor; no teniendo más nada que observar.

De conformidad a la normativa aplicable firmamos, en Riobamba 03 días del mes de octubre del 2023.

Lady Espinoza, Mgs.
PRESIDENTE DEL TRIBUNAL DE GRADO



Ana Congacha, Mgs.
MIEMBRO DEL TRIBUNAL DE GRADO



Hugo Paz, Mgs.
MIEMBRO DEL TRIBUNAL DE GRADO



CERTIFICADO ANTIPLAGIO



Dirección
Académica
VICERRECTORADO ACADÉMICO



CERTIFICACIÓN

Que, **Shiguango García, Jonathan Cristoffer** con CC: 1500782162 y **Muñoz Sarmiento, Anderson Joel** con CC: 1726009101, estudiantes de la Carrera **Ingeniería en Tecnologías de la Información, VIGENTE**, Facultad de **Ingeniería**; han trabajado bajo mi tutoría el trabajo de Investigación titulado "**APLICACIÓN WEB Y MÓVIL DE VIGILANCIA PARA EL CONSUMO DE MEDICAMENTOS EN PACIENTES AMBULATORIOS, UTILIZANDO "FOG COMPUTING"**", cumple con el 4 %, de acuerdo al reporte del sistema Anti plagio **URKUND**, porcentaje aceptado de acuerdo a la reglamentación Institucional, por consiguiente autorizo continuar con el proceso.

Riobamba, 21 de septiembre de 2023



MGS. JORGE DELGADO
ALTAMIRANO

Mgs. Jorge Delgado
TUTOR(A)

DEDICATORIA

Dedico este trabajo de investigación principalmente a Dios que es el motor fundamental de toda mi vida por consiguiente a mis padres Jorge y Margarita que han estado durante todo este trayecto en mi vida tanto apoyo moral como económico han sido las personas que en todo momento bueno o malo han sabido entenderme y apoyar, esta etapa de mi vida culmina de manera satisfactoria y le doy nuevamente gracias a Dios por este logro que servirá para sustentar una familia y romper estereotipos de muchas clases.

Jonathan Cristoffer Shiguango Garcia

Dedico este trabajo de investigación principalmente a Dios, por ser el inspirador y darnos fuerza para continuar en este proceso de obtener uno de los anhelos más deseados. A mis padres, por su amor, trabajo y sacrificio en todos estos años, gracias a ustedes he logrado llegar hasta aquí y convertirme en lo que soy. Ha sido el orgullo y privilegio el ser su hijo, son los mejores padres.

Anderson Joel Muñoz Sarmiento

AGRADECIMIENTO

En primer lugar, doy gracias a Dios quien es el ente que guía mi vida y ha sido la fortaleza para poder salir adelante en cada obstáculo que en esta vida estudiantil he atravesado y por poner en mi camino a personas tan grandes de corazón.

A las personas que más estimo mis padres Jorge y Margarita que han sido base fundamental en todos los aspectos que se pueda imaginar, les agradezco que han podido ayudarme a pesar de todas las falencias que yo pude ocasionar.

De igual forma debo agradecer a la Universidad Nacional de Chimborazo por los grandes docentes que tiene, han sido los que me han instruido a seguir mi vocación y conseguir grandes logros académicos.

A mi jurado calificador que han sido docentes que me han conocido y han hecho de mi un profesional el día de hoy les doy gracias y bendiciones.

Por último, pero no menos importante a mi docente tutor Mg. Jorge Edwin Delgado Altamirano que gracias a su guía he podido culminar un escalón más y seguir profesionalizándome para conseguir más metas en todos los ámbitos posibles.

El camino que he seguido ha sido duro, pero gracias a todas las personas anteriormente mencionadas hoy puedo decir lo logre y seguiré mi camino como los profesionales que me formaron mil gracias.

Jonathan Cristoffer Shiguango Garcia

AGRADECIMIENTO

En primer lugar, les agradezco a mis padres que siempre me han brindado su apoyo incondicional para poder cumplir todos mis objetivos personales y académicos. Ellos son los que con su cariño me han impulsado siempre a perseguir mis metas y nunca abandonarlas frente a las adversidades. También son los que me han brindado el soporte material y económico para poder concentrarme en los estudios y nunca abandonarlos.

Le agradezco muy profundamente a mi tutor por su dedicación y paciencia, sin sus palabras y correcciones precisas no hubiese podido lograr llegar a esta instancia tan anhelada. Gracias por su guía y todos sus consejos, los llevaré grabados para siempre en la memoria en mi futuro profesional.

Son muchos los docentes que han sido parte de mi camino universitario, y a todos ellos les quiero agradecer por transmitirme los conocimientos necesarios para hoy poder estar aquí. Sin ustedes los conceptos serían solo palabras, y las palabras ya sabemos quién se las lleva, el viento.

Agradecerles a todos mis compañeros los cuales muchos de ellos se han convertido en mis amigos, cómplices y hermanos. Gracias por las horas compartidas, los trabajos realizados en conjunto y las historias vividas

Por último, agradecer a la universidad que me ha exigido tanto, pero al mismo tiempo me ha permitido obtener mi tan ansiado título. Agradezco a cada directivo por su trabajo y por su gestión, sin lo cual no estarían las bases ni las condiciones para aprender conocimientos.

Anderson Joel Muñoz Sarmiento

ÍNDICE GENERAL

DECLARATORIA DE AUTORÍA.....	
DICTAMEN FAVORABLE DEL PROFESOR TUTOR	
CERTIFICADO DE LOS MIEMBROS DEL TRIBUNAL	
CERTIFICADO ANTIPLAGIO.....	
DEDICATORIA.....	
AGRADECIMIENTO	
1. CAPÍTULO I. INTRODUCCION.....	16
PLANTEAMIENTO DEL PROBLEMA	18
1.1 Problema y Justificación	18
1.2 Formulación del problema	18
1.3 Objetivos	19
1.3.1 General	19
1.3.2 Específicos.....	19
2. CAPÍTULO II. MARCO TEÓRICO.....	20
2.1 Antecedentes	20
2.2 Arquitectura de IoT.....	20
2.3 Aplicaciones del IoT en las Mhealth.....	22
2.4 Cloud Computing.....	23
2.4.1 Arquitectura cloud computing	23
2.5 Fog computing	25
2.5.1 Características de fog computing.....	26
2.5.2 Arquitectura fog computing.....	27
2.6 Aplicativos web y móviles	28
2.6.1 Android Studio	29
2.6.2 Visual Studio Code	30
2.6.3 Lenguajes de programación	31
2.7 Virtualización.....	34
2.7.1 Plataformas de Virtualización.....	34
2.7.2 Sistema Operativo Windows Server 2016	35

2.8	Base de Datos PhpMyAdmin	36
2.9	Metodologías ágiles	36
2.9.1	Metodología XP.....	37
2.10	Norma ISO 25000	37
2.10.1	Eficiencia de desempeño (Rendimiento).....	38
3.	CAPÍTULO III. METODOLOGIA.....	38
3.1	Metodología	38
3.2	Tipo y diseño de la investigación.....	38
3.2.1	Según la fuente de investigación	38
3.2.2	Según el objeto de estudio	38
3.2.3	Según el tipo de variable	39
3.3	Unidad de Análisis.....	39
3.3.1	Población y muestra.....	39
3.4	Entrevistas.....	39
3.5	Encuestas	39
3.6	Técnicas de análisis e interpretación de la información	39
3.7	Desarrollo aplicativo web y móvil	40
3.7.1	Fase de Exploración.....	41
3.7.2	Fase de Planificación	42
3.7.3	Diseño del aplicativo web.....	44
3.7.4	Diseño del aplicativo móvil	46
3.7.5	Diseño arquitectónico	48
3.7.6	Diseño de la infraestructura de fog computing	49
3.8	Desarrollo.....	50
3.8.1	Desarrollo del aplicativo web	50
3.8.2	Base de datos	54
3.8.3	Desarrollo del aplicativo móvil	57
4.	CAPÍTULO IV. RESULTADOS Y DISCUSIÓN	61
4.1	Resultado de pruebas	61
4.1.1	Pruebas unitarias.....	61
4.1.2	Pruebas de aceptación.....	63

4.1.3	Detección de errores	64
4.1.4	Rendimiento	64
4.1.5	Evaluación del sistema usando jmeter	66
4.2	Resultados de encuestas	71
5.	CAPÍTULO V. CONCLUSIONES Y RECOMENDACIONES	75
5.1	Conclusiones	75
5.2	Recomendaciones	76
	BIBLIOGRAFÍA.....	77
	ANEXOS.....	80

ÍNDICE DE TABLAS.

<i>Tabla 1:</i>	<i>Plataformas de desarrollo y fog computing</i>	<i>28</i>
<i>Tabla 2:</i>	<i>Comparativa de máquinas virtuales y compatibilidad fog computing</i>	<i>34</i>
<i>Tabla 3:</i>	<i>Estructura de la planificación</i>	<i>42</i>
<i>Tabla 4:</i>	<i>Descripción de variables modulo Recetas.....</i>	<i>52</i>
<i>Tabla 5:</i>	<i>Descripción de variables Medicamento</i>	<i>53</i>
<i>Tabla 6:</i>	<i>Descripción de variables de Pacientes</i>	<i>54</i>
<i>Tabla 7:</i>	<i>Estructura en PHPmyadmin de entidad Medico</i>	<i>54</i>
<i>Tabla 8:</i>	<i>Estructura en PHPmyadmin de entidad Paciente</i>	<i>55</i>
<i>Tabla 9:</i>	<i>Estructura en PHPmyadmin de entidad Medicamentos</i>	<i>55</i>
<i>Tabla 10:</i>	<i>Estructura en PHPmyadmin de entidad Receta</i>	<i>56</i>
<i>Tabla 11:</i>	<i>Tiempos de respuesta tabulado en 100 clientes.</i>	<i>68</i>
<i>Tabla 12:</i>	<i>Tiempos de respuesta tabulado en 250 clientes.</i>	<i>68</i>

Tabla 13: <i>Tiempos de respuesta tabulado en 500 clientes</i>	69
Tabla 14: <i>Tiempos de respuesta tabulado</i>	70

ÍNDICE DE FIGURAS

Figura 1: <i>Arquitectura IoT</i>	21
Figura 2: <i>Salud móvil</i>	23
Figura 3: <i>Software como servicio</i>	24
Figura 4: <i>Plataforma como servicio</i>	25
Figura 5: <i>Infraestructura como servicio</i>	25
Figura 6: <i>Arquitectura fog computing</i>	28
Figura 7: <i>Interfaz Android Studio</i>	30
Figura 8: <i>Visual Studio Code</i>	31
Figura 9: <i>Lenguajes de programación</i>	31
Figura 10: <i>Ejemplo de Código Java</i>	32
Figura 11: <i>Ejemplo PHP "Hola Mundo"</i>	33
Figura 12: <i>Uso de HTML</i>	33
Figura 13: <i>Código XML</i>	34
Figura 14: <i>Interfaz PhpMyAdmin</i>	36

Figura 15: Estructura programación Xp.....	37
Figura 16: Normas de calidad del producto software.....	37
Figura 17: Herramientas de desarrollo web y móvil	40
Figura 18: Encuesta a los pacientes de la comunidad Sindy	41
Figura 19: Resultado de entrevista a Doctor Elder	42
Figura 20: Arquitectura sistema web	48
Figura 21: Arquitectura aplicativo móvil	49
Figura 22: Arquitectura fog computing.....	49
Figura 23: Arquitectura de fog computing integrada.....	50
Figura 24: Código HTML y CSS en Visual Studio Code.....	51
Figura 25: Modulo Receta	51
Figura 26: Modulo Medicamentos	52
Figura 27: Modulo Pacientes	53
Figura 28: Diagrama E_R de la base de datos	57
Figura 29: Interfaz de desarrollo de splash	58
Figura 30: Interfaz de desarrollo de login	58
Figura 31: Interfaz de desarrollo de menú principal	59
Figura 32: Interfaz de desarrollo de perfil de paciente.....	59
Figura 33: Interfaz de desarrollo de medicamentos.....	60
Figura 34: Conectividad con smartphone	61
Figura 35: Conectividad con computadora.....	62
Figura 36: Verificación de información	62
Figura 37: Verificación de información del paciente.....	63
Figura 38: Ejecución del sistema	63
Figura 39: Envío de ping	64
Figura 40: Panel de AIDA64	65
Figura 41: Ejecución de panel de AIDA64.....	65
Figura 42: Prueba de Jmeter con 100 clientes.....	66
Figura 43: Prueba de Jmeter con 250 clientes.....	67
Figura 44: Prueba de Jmeter con 500 clientes.....	67

RESUMEN

Durante la pandemia de COVID-19, se evidenció la falta de sistemas de vigilancia en la toma de medicamentos, especialmente entre los grupos más vulnerables, como adultos mayores, niños y personas con enfermedades catastróficas. Muchos de ellos no seguían adecuadamente las prescripciones médicas en términos de horarios y dosis. Para abordar esta problemática, se propuso desarrollar un aplicativo móvil llamado medycal system, integrado en un sistema de vigilancia para pacientes ambulatorios.

Este aplicativo proporciona ayuda tanto a los pacientes como a los médicos al recordar los horarios de toma de medicamentos. Además, el médico puede obtener información sobre si el paciente cumple con las pautas de medicación, lo que se incorpora a su historial médico y se revisa mediante una interfaz web del sistema.

Para lograr una comunicación eficiente, se creó un entorno de red LAN que emplea la tecnología de fog computing. Un servidor centralizado suministra toda la información al sistema de vigilancia en la toma de medicamentos. Se ha observado una velocidad de transmisión aceptable de la información de las tomas realizadas por los pacientes hacia el gestor de tomas web. Esta implementación se alinea con las ventajas encontradas en artículos científicos investigados durante el desarrollo del proyecto, como una conexión estable y eficiente para el sistema.

El uso de la tecnología de fog computing en dispositivos que interactúan en el ecosistema del internet de las cosas (IoT), con enfoque en la salud, permite prescindir de una conexión

constante a una red Cloud, evitando problemas de latencia y aprovechando una conexión a un servidor local. Esto beneficia en términos de velocidad a la hora de cumplir con una toma de medicamento asignada.

Palabras claves: fog computing, aplicativo movil, sistema web, pacientes ambulatorios, IoT, Rendimiento

ABSTRACT

During the COVID-19 pandemic, the lack of medication monitoring systems became evident, especially among the most vulnerable groups, such as older adults, children, and individuals with catastrophic illnesses. Many needed to adhere to medical prescriptions regarding timing and dosage correctly. To address this issue, a mobile application called "medical system" was proposed for development, integrated into an outpatient medication monitoring system.

This application assists both patients and doctors by reminding them of medication schedules. Additionally, the doctor can obtain information about whether the patient is following medication guidelines, which are incorporated into their medical history and reviewed through the system's web interface. A LAN network environment was created using fog computing technology to achieve efficient communication. A centralized server supplies all the information to the medication monitoring system. An acceptable transmission speed of the information regarding medication doses patients take to the web-based dose manager has been observed. This implementation aligns with the advantages found in scientific articles researched during the project's development, such as a stable and efficient connection for the system.

The use of fog computing technology in devices that interact within the Internet of Things (IoT) ecosystem, with a focus on healthcare, permits the avoidance of a constant connection to a cloud network, thereby avoiding latency issues and leveraging a connection to a local server. This benefits in terms of speed when adhering to assigned medication dose.

Keywords: fog computing, mobile application, web system, outpatient patients, IoT, performance.



Reviewed by:
Mgs. Lorena Solís Viteri
ENGLISH PROFESSOR
c.c. 0603356783

1. CAPÍTULO I. INTRODUCCION.

En la actualidad, el internet de las cosas se ha transformado en base fundamental de la comunicación entre dispositivos tecnológicos como: drones, sensores, smartphones, vehículos, actuadores, servidores, laptops, entre otros, los mismos que pueden ser monitoreados y conectados. Por el crecimiento del internet de las cosas el uso de recursos informáticos y de almacenamiento ha aumentado de manera considerable por consecuencia se ha abierto diferentes estudios sobre donde deberían ser alojadas las diferentes datas, el concepto cloud computing enfatiza la tecnología que permite un acceso remoto a gestión de archivos, software y control remoto mediante una conexión a internet.

El alto requerimiento que ocupa el internet de las cosas hace que la arquitectura de cloud computing no sea la opción más adecuada ya que el principal objetivo es poder dar una interconectividad de aplicaciones de manera más eficiente y rápida, además cloud computing usa servidores que están muy lejos del usuario esto implica que exista un retardo y problemas de latencia en el envío de información que es su principal función, la terminología fog computing involucra la mejora en el procesamiento de datos en lo que tales pueden ser procesados de mejor manera en un servidor local cercano al usuario, de igual forma procesan datos de dispositivos locales que no se encuentran inmersos en el internet sino que estén lo más cerca posible, de esa forma mejoran los tiempos de respuestas, por ende proporcionan almacenamiento y procesamiento de datos de IoT localmente en vez de enviarlos a la nube.

El internet de las cosas está involucrado en las diferentes áreas de turismo, educación, comercio, industria, salud, entre otros siendo este último el sector en el que se desarrollan múltiples aplicativos móviles de asistencia ambulatoria para ello existe Mhealth que no es más que práctica de la medicina y la salud pública respaldada por dispositivos móviles como teléfonos, monitores de pacientes, asistentes digitales y otros dispositivos inalámbricos, Mhealth ayuda a la interacción entre dispositivos móviles, sensores y las personas para el monitoreo de alguna variable que se relaciona con la salud esto hace posible que la interacción de la asistencia ambulatoria sea en tiempo real.

La asistencia ambulatoria a diversos pacientes requiere una implementación de una arquitectura y servicios en tiempo real que permitan el control y vigilancia, en Ecuador debido a la falta de aplicativos de monitoreo los diferentes pacientes no logran tomar a

tiempo los medicamentos que el doctor ha enviado mediante receta puede ser por factores de incapacidad física, intelectual u otra, para ello se plantea en la carrera de TI de la UNACH se desarrolle un aplicativo móvil y web con tecnología fog computing que permita el control y vigilancia en tiempo real de pacientes ambulatorios el cual permitirá salvaguardar la vida del paciente.

Después de todo lo antes mencionado este trabajo de investigación se propone a la realización de una aplicación web y móvil para la salud ambulatoria para que este ayude a que sea monitoreada en tiempo real con el objetivo de ayudar a los pacientes a la toma de medicamentos utilizando el software Android Studio virtual box y el lenguaje JAVA, HTML y PHP.

La estructuración del proyecto de investigación es: En el capítulo I describe el planteamiento del problema, formulación del problema y objetivos, en el capítulo II se menciona el marco teórico que permite conocer los conceptos relacionados con aplicaciones móviles, aplicaciones web, cloud computing, fog computing, internet de las cosas, Mhealth, capítulo III se indica la metodología usada y en el capítulo IV se representa los resultados obtenidos de la evaluación de la eficiencia de desempeño de la aplicación móvil utilizando el criterio utilización de recursos de rendimiento, capítulo V se menciona las conclusiones y recomendaciones.

PLANTEAMIENTO DEL PROBLEMA

1.1 Problema y Justificación

Según Rodríguez (2019) los pacientes con alta vulnerabilidad respecto a la salud son niños/niñas, adultos mayores, personas con enfermedades crónicas. a partir del diagnóstico del médico los pacientes reciben tratamiento mediante la prescripción de medicamentos que se deben aplicar en ciertos periodos de tiempo y en cantidades específicas, una de las mayores dificultades en los pacientes es el control y vigilancia en la aplicación de medicamentos en los horarios establecido por el médico.

Los dispositivos móviles, Mhealth y la arquitectura fog computing están cambiando muchos aspectos en la sociedad es por ello hoy en día es posible que desde una aplicación móvil Mhealth se puede realizar notificaciones para la toma correcta de medicamentos de los pacientes en donde se puede incluir diferentes personas como son médico, paciente y familiar por el amplio circulo social en el que vivimos, existen personas con enfermedades crónicas que necesitan de manera oportuna una atención priorizada y en el peor de los casos una atención hospitalaria inmediata todo esto lo hace posible el fog computing el cual posee un servidor local que en zonas que el cloud computing no llega de manera inmediata hace posible y garantiza la comunicación en tiempo real de cualquier información relevante.

1.2 Formulación del problema

¿Cómo el rendimiento de la Aplicación Web y Móvil incidirá en el control y vigilancia en el consumo de medicamentos en pacientes ambulatorios?

1.3 Objetivos

1.3.1 General

Desarrollar una Aplicación Web y Móvil de vigilancia para el consumo de medicamentos en pacientes ambulatorios, utilizando fog computing.

1.3.2 Específicos

- Investigar la tecnología y servicios de fog computing para asistencia ambulatoria en el ámbito de la salud
- Implementar una Aplicación Web y Móvil de vigilancia para el consumo de medicamentos en pacientes ambulatorios, utilizando fog computing.
- Evaluar el rendimiento de la aplicación web de vigilancia para el consumo de medicamentos en pacientes ambulatorios, utilizando Jmeter.

2. CAPÍTULO II. MARCO TEÓRICO.

2.1 Antecedentes

El Internet de las cosas (IoT) es un concepto que se refiere a la interconexión de diversos dispositivos y objetos a través de una red (ya sea privada o pública), permitiendo que todos estos elementos sean visibles e interactúen entre sí. En el ámbito del IoT, una amplia variedad de dispositivos, desde sensores y dispositivos mecánicos hasta objetos cotidianos como refrigeradores, calzado inteligente, smartwatches, entre otros, pueden estar conectados a Internet y comunicarse sin necesidad de intervención humana. El objetivo principal del IoT es facilitar la comunicación máquina a máquina (M2M) o dispositivos M2M, lo que permite la automatización y la toma de decisiones inteligentes sin una intervención manual constante (María Gracia, 2022).

Actualmente, la infraestructura tecnológica en áreas urbanas está bien desarrollada y permite el acceso a redes de alta potencia. Sin embargo, las empresas ubicadas en entornos rurales a menudo enfrentan el desafío de no tener acceso a redes de alta velocidad y gran ancho de banda. Esto se convierte en un problema cuando se busca implementar conceptos como la fábrica inteligente (Industria 4.0), donde una planta industrial completamente conectada puede generar grandes volúmenes de información diariamente, alcanzando centenares de gigabytes (Digital Guide Ionos, 2019).

2.2 Arquitectura de IoT

En la **figura 1** se muestra todos los componentes de IoT que hacen posible el envío de datos entre ellas a continuación su estructura o arquitectura:

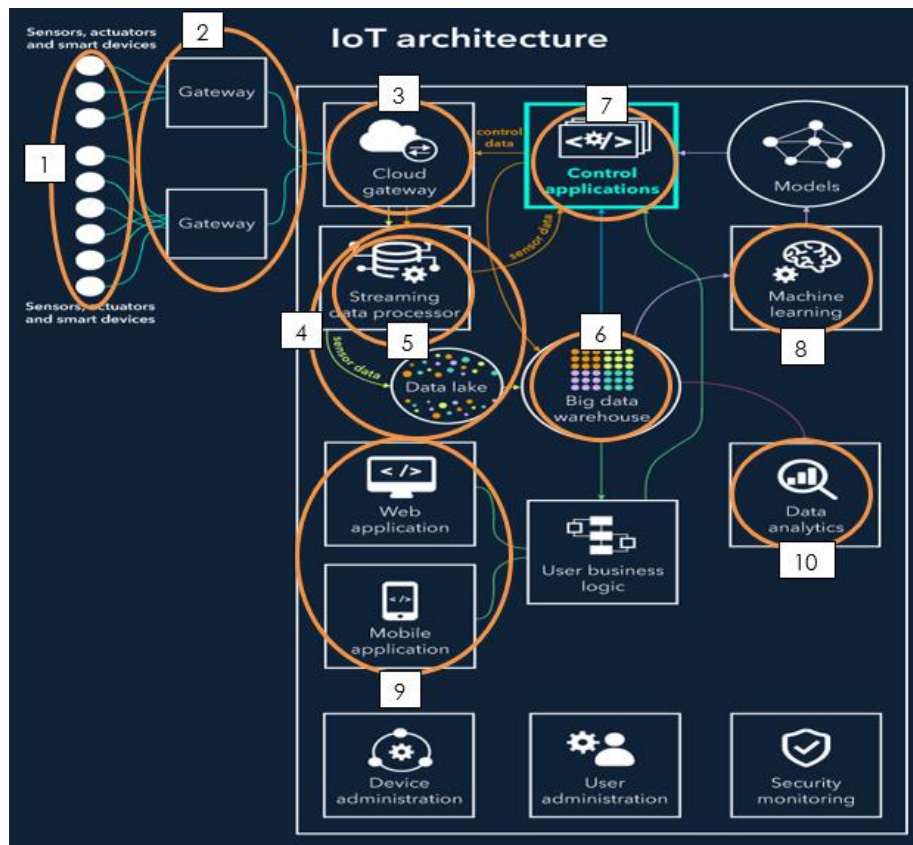


Figura 1: Arquitectura IoT

- **1.-** Cosas equipadas con **sensores** para recoger datos y actuadores para realizar comandos recibidos desde la nube.
- **2.-Gateways** para filtrar, preprocesar y mover datos a la nube y viceversa.
- **3.-** Pasarelas en nube (**Cloud Gateways**) para garantizar la transición de datos entre las pasarelas sobre el terreno y los servidores centrales de IoT.
- **4.-Procesadores de datos** en tiempo real para distribuir los datos procedentes de los sensores entre los componentes de las soluciones de IoT pertinentes.
- **5.-Bases de Datos** para almacenar todos los datos de valor definido e indefinido.
- **6.-Big Data Warehouse** para la recogida de datos valiosos.
- **7.- Aplicaciones de control** para enviar comandos a los actuadores.
- **8.-Machine Learning** para generar los modelos que serán después utilizados por las aplicaciones de control.
- **9.-Aplicaciones de usuario** ayuda a los usuarios a monitorear los dispositivos conectados.
- **10.-Análisis de datos** para un correcto procesamiento de los datos a información.

2.3 Aplicaciones del IoT en las Mhealth

La salud asistida por el móvil o MHealth (Mobile Health) se refiere al uso de dispositivos móviles, como smartphones, tablets y otros dispositivos portátiles, para brindar servicios médicos y sanitarios. Esta tecnología ha abierto nuevas oportunidades en el campo de la atención médica, permitiendo a los profesionales de la salud y a los pacientes acceder a información relevante, realizar seguimiento de condiciones médicas y mejorar la comunicación entre ellos de manera más rápida y efectiva (Team, 2021).

En la sociedad actual el uso de IoT en los sistemas de Mhealth se ha vuelto tendencia por el alto requerimiento de envío y recepción de información entre varios componentes médicos, dispositivos móviles y otros ordenadores.

La incorporación de las (IoT) permite multiprocesos como pueden ser:

Conectividad de dispositivos y equipos médicos: Conectan de eficiente manera a personas y grupos para mejorar el nivel de diagnóstico y una rápida atención.

Monitoreo remoto: Acopla datos médicos y de salud de muchas personas en un solo lugar para transmitir de manera segura la información y de forma electrónica a los proveedores de atención medica en otro lugar para que los puedan procesar y dar un diagnóstico efectivo y rápido.

Seguimiento ambulatorio: Aprovecha las características de las (IoT) para que hospitales, médicos de emergencia, y droguerías monitoreen de manera efectiva a los pacientes en diversas condiciones que tenga el paciente relacionado a la salud.

Iot salud.

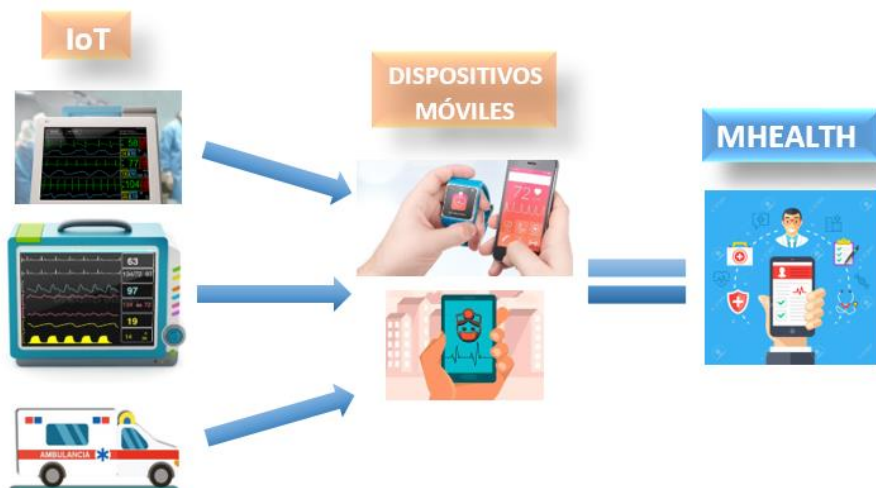


Figura 2: Salud móvil

En **figura 2** se muestra el intercambio de datos que existe desde las máquinas de monitoreo hacia los dispositivos móviles, desde el manejo de historiales clínicos hasta la medición de signos vitales del paciente.

2.4 Cloud Computing

Así es, la computación en la nube ha revolucionado la forma en que utilizamos la tecnología de la información (TI) y se ha convertido en un modelo comercial fundamental para la entrega de recursos de TI. Con la computación en la nube, las personas y las organizaciones pueden acceder a una variedad de recursos informáticos a través de la red de manera rápida y bajo demanda. Estos recursos incluyen servidores, almacenamiento y aplicaciones, entre otros (Sunyaev, 2020).

La principal característica de la computación en la nube es la capacidad de compartir y acceder a estos recursos de forma escalable, lo que significa que se pueden asignar y desasignar recursos según sea necesario, lo que permite una mayor flexibilidad y eficiencia en el uso de la infraestructura informática (Sunyaev, 2020).

2.4.1 Arquitectura cloud computing

La computación en la nube está basada en tres capas fundamentales tales como:

2.4.1.1 Software como Servicio (SaaS)

En la **figura 3** se observa conexión aplicación completa como un servicio. El proveedor SaaS dispone de una aplicación estándar desarrollada en algunos casos por la misma persona que se encarga de operar y mantener y con la que da servicio a multitud de clientes a través de la red, sin que estos tengan que instalar ningún software adicional así los costes se reducen enormemente. Un ejemplo claro es la aplicación para el manejo del correo electrónico como Gmail por medio de un web-browser (Torres, 2022).

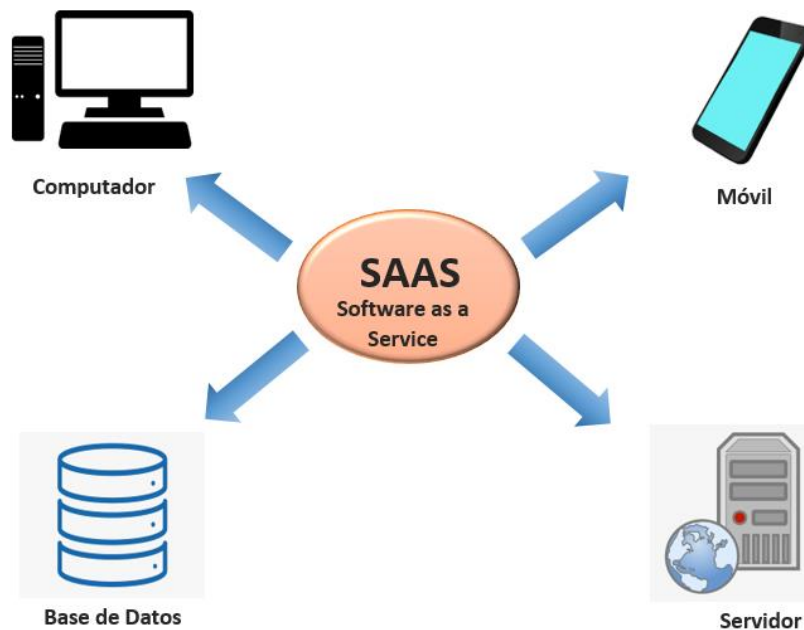


Figura 3: Software como servicio

2.4.1.2 Plataforma como Servicio (PaaS)

En la **figura 4** se muestra una plataforma como servicio se enfoca hacia aplicaciones multimodulares o hacia el despliegue de soluciones de infraestructuras más complejas. Para lograr esto generalmente usa la orquestación como medio para conectar componentes de software y hardware para ofrecer una solución completa al cliente sin necesidad de largos procesos de despliegue. Ej. Sistemas Crm, SharePoint, Exchange etc. El usuario usa la solución, paga por usarla y el proveedor se encarga de la automatización y el despliegue de dichas aplicaciones (hostingred, 2022).

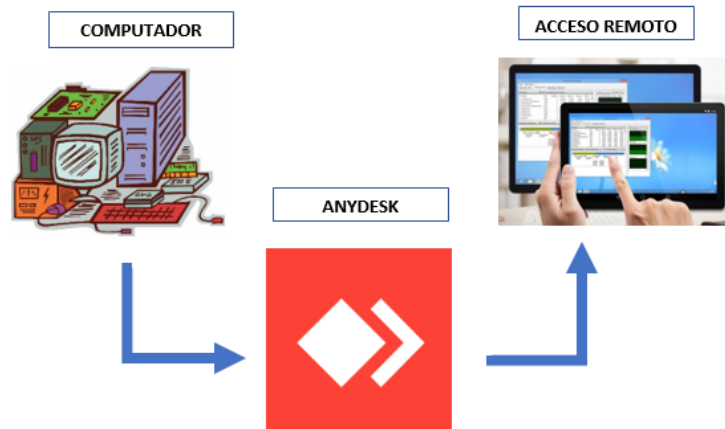


Figura 4: Plataforma como servicio

2.4.1.3 Infraestructura como Servicio (IaaS)

En la **figura 5** se muestra IaaS (Infraestructura como Servicio) es un modelo de cloud computing que ofrece a las empresas una solución para obtener y gestionar recursos de computación a través de Internet. Con IaaS, los proveedores de servicios en la nube suministran y gestionan recursos esenciales, como servidores, almacenamiento, redes y virtualización, permitiendo a las empresas acceder a una infraestructura de alto nivel sin necesidad de realizar una inversión significativa en equipamiento de TI (Stackscala, 2021).

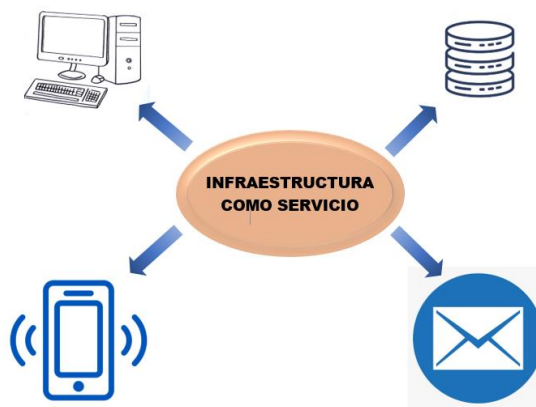


Figura 5: Infraestructura como servicio

2.5 Fog computing

La computación en la niebla (fog computing) es un concepto que fue introducido por la empresa Cisco en el año 2012. Según la definición de Cisco, la niebla extiende la nube más

cerca de los dispositivos y objetos que generan y actúan sobre los datos del Internet de las cosas (IoT). Esta tecnología se basa en la idea de distribuir la capacidad de cómputo y almacenamiento más cerca del borde de la red, es decir, más cerca de los dispositivos IoT, en lugar de centralizar todo el procesamiento en la nube (Perez, 2021).

Los dispositivos que implementan la computación en la niebla se conocen como "nodos niebla". Estos nodos pueden ubicarse en cualquier lugar donde haya una conexión de red, como en una planta de fábrica, en la parte superior de un poste de energía, junto a una vía férrea, en un vehículo o en una plataforma petrolera, entre otros. Cualquier dispositivo que tenga capacidad de cómputo, almacenamiento y conectividad de red puede ser un nodo niebla. Algunos ejemplos de dispositivos que pueden ser nodos niebla incluyen controladores industriales, conmutadores, enrutadores, servidores integrados y cámaras de videovigilancia (Perez, 2021).

2.5.1 Características de fog computing

El Fog computing, al igual que otros modelos de informática basada en IoT o TI, presenta características distintivas. Aquí analizaremos tres de ellas:

1. **Minimización de la latencia y conocimiento de la ubicación:** En el contexto del fog computing, los procesos se ejecutan en proximidad a la fuente de los datos o directamente en el lugar donde se originan. Este enfoque reduce significativamente los tiempos necesarios para analizar y procesar la información, lo que da como resultado respuestas rápidas y precisas en el momento justo que se requieren. Esto, a su vez, facilita la toma de decisiones ágiles y pertinentes.

2. **Amplia presencia de nodos:** Los nodos de niebla, también conocidos como nodos de borde, desempeñan un papel esencial en el fog computing. Funcionan como puntos intermedios inteligentes de conexión entre los dispositivos finales y la infraestructura en la nube. Estos nodos de niebla tienen la capacidad de tomar decisiones autónomas sobre la descentralización del procesamiento de datos, determinando qué información debe procesarse localmente y cuál debe ser enviada a los centros de datos en la nube central. Este enfoque se alinea con los objetivos clave del fog computing, que son acortar las rutas de comunicación y reducir la carga de datos en la nube.

3. Independencia de la conectividad a Internet: Otra característica importante del fog computing es su capacidad para funcionar sin depender exclusivamente de una conexión a Internet. Dado que gran parte de los procesos se realizan en la red local o en dispositivos cercanos a la fuente de datos, no se requiere acceso constante a la nube para la captura y el procesamiento de la información. Esto permite que las aplicaciones y programas continúen operando incluso en ausencia de conectividad, con la posibilidad de actualizarlos cuando la conexión se restablece.

2.5.2 Arquitectura fog computing

En la **figura 6** se muestra la infraestructura de fog computing que suele dividirse en 3 capas como son:

Edge Layer: La capa de borde incluye a todos los dispositivos inteligentes, es decir, aquellos situados en el límite de la red, en el contexto de una arquitectura de IoT. Los datos generados en esta capa pueden ser procesados en el propio dispositivo o transmitidos a un servidor, denominado nodo fog, que se encuentra en la capa de niebla.

Fog Layer: La capa de niebla está compuesta por una serie de servidores altamente eficientes que reciben los datos de la capa inicial, los adecuan y, en caso de requerirse, los envían hacia la nube.

Cloud Layer: La capa en la nube marca el punto de culminación de una estructura de computación en la niebla.

La plataforma en la nube almacena la información de la producción para diferentes aplicaciones de ingeniería, que son publicadas y ejecutadas por nodos de niebla desplegados dentro de la red local de las instalaciones. Estos nodos locales ofrecen seguridad y privacidad de datos, estas comunicaciones dependen de las políticas y servicios de seguridad existentes en las instalaciones (Caiza, 2020).



Figura 6: Arquitectura fog computing

2.6 Aplicativos web y móviles

Una aplicación web, también conocida como aplicación web o aplicación en línea, es una aplicación que se ejecuta en un navegador web y no requiere ser instalada en el dispositivo del usuario. Está compuesta principalmente por HTML (lenguaje de marcado), JavaScript (lenguaje de programación) y CSS (hojas de estilo en cascada). Estos tres elementos trabajan juntos para crear la interfaz de usuario y la funcionalidad de la aplicación (Desarrollo Web, 2019).

Las aplicaciones móviles se crean para ofrecer servicios y experiencias a los usuarios en sus dispositivos móviles de una manera optimizada y centrada en la movilidad. Aunque las aplicaciones móviles suelen ser más pequeñas en comparación con las aplicaciones de escritorio, pueden proporcionar una amplia variedad de funciones y características, desde simples herramientas hasta aplicaciones complejas con múltiples funcionalidades (Anincubator, 2021).

A continuación, se mostrará el cuadro comparativo de plataformas de desarrollo con fog computing,

Tabla 1: Plataformas de desarrollo y fog computing

Características	Utilidad con fog computing
Android Studio	
<ul style="list-style-type: none"> • Sistema de compilación flexible basado en Gradle • Emulador rápido y cargado de funciones • Entorno unificado donde puedes desarrollar para todos los dispositivos Android • Aplicación de cambios para insertar cambios de código y recursos a la app en ejecución sin reiniciarla • Integración con GitHub • plantillas de código para ayudar a compilar funciones de apps • Variedad de marcos de trabajo y herramientas de prueba 	<ul style="list-style-type: none"> • Esta plataforma es de fácil uso y sencilla • Herramientas disponibles • Compatibilidad • Servicios colgados en internet
Vuforia Augmented Reality	
<ul style="list-style-type: none"> • Permite promover la realidad aumentada con objetos 3D del mundo real 	<ul style="list-style-type: none"> • Plataforma que desarrolla entornos en 3D. • Compatible con fog computing
AppMachine	
<ul style="list-style-type: none"> • Permite un desarrollo fácil de usar ya que admite el diseño y desarrollo de aplicaciones nativas para Android e iOS. Los desarrolladores solo necesitan incluir componentes básicos en la funcionalidad, incluidas fotos y videos. 	<ul style="list-style-type: none"> • Su compatibilidad con fog computing • Es óptima cuando el objetivo de la app móvil sea un aplicativo de diseño simple que no incluya elemento multimedia
Gimbal Context-Aware	
<ul style="list-style-type: none"> • Permite configurar fácilmente geocercas para definir puntos de acceso en las interacciones dentro de la aplicación. El cardán también monitorea la actividad de la red y las preferencias del usuario para personalizar el rendimiento de la aplicación para cada usuario. 	<ul style="list-style-type: none"> • Su compatibilidad con fog computing • Óptima en el caso de aplicativos que usen delimitaciones geográficas virtuales

Fuente: (Shiguango-Muñoz, 2023)

2.6.1 Android Studio

En la **figura 7** se muestra la plataforma de desarrollo más óptima que denominada Android Studio es: el entorno de desarrollo integrado (IDE) oficial para el desarrollo de apps para

Android y está basado en IntelliJ IDEA. Además del potente editor de códigos y las herramientas para desarrolladores de IntelliJ, Android Studio ofrece incluso más funciones que aumentan tu productividad además tiene un sistema de compilación flexible, emulación rápido, entorno unificado, compatibilidad con C++ y NDK, variedad de marcos, etc.

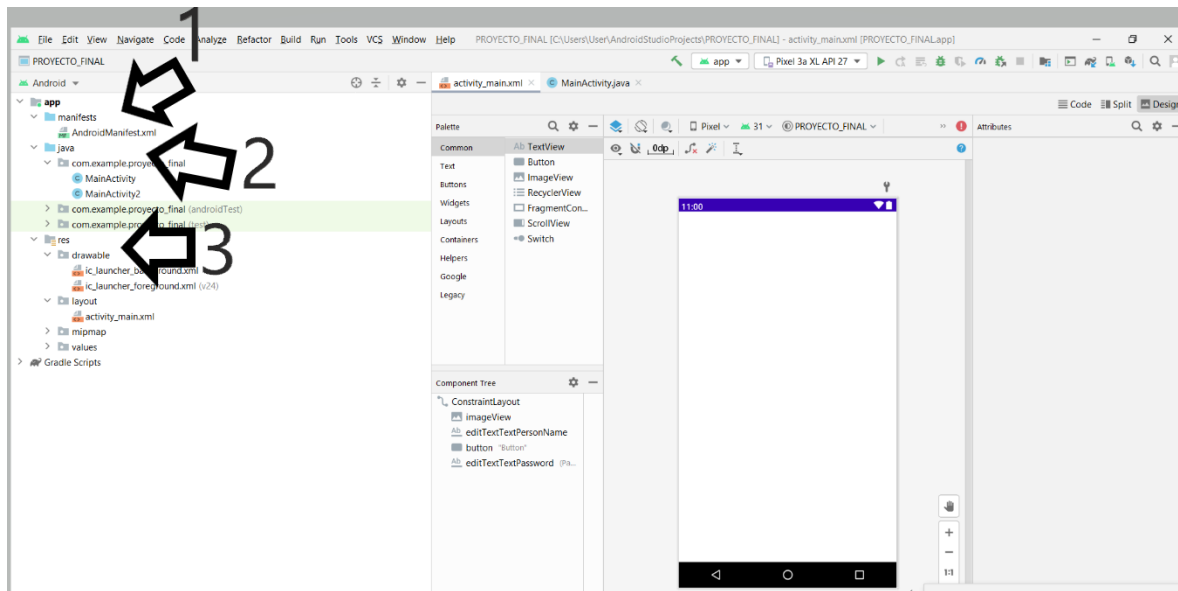


Figura 7: Interfaz Android Studio

En la **figura 7** se puede observar:

- 1: **manifests:** contiene el archivo AndroidManifest.xml.
- 2: **java:** contiene los archivos de código fuente Java, incluido el código de prueba de JUnit.
- 3: **res:** contiene todos los recursos sin código, como diseños XML, strings de IU e imágenes de mapa de bits (developers, 2018).

2.6.2 Visual Studio Code

En la **figura 8** se muestra el editor de texto plano desarrollado por Microsoft totalmente gratuito y de código abierto para ofrecer a los usuarios una herramienta de programación avanzada como alternativa al Bloc de Notas (Velasco, 2021).

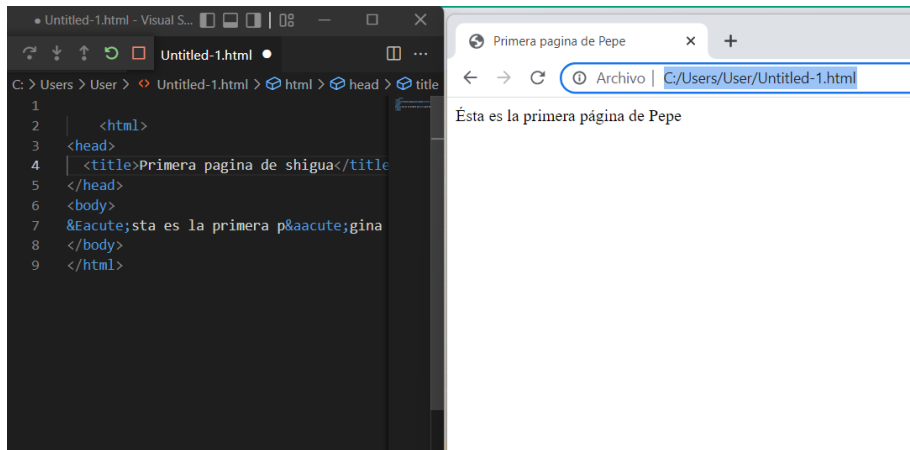


Figura 8: Visual Studio Code

2.6.3 Lenguajes de programación

Es un programa destinado a la construcción de otros programas informáticos. Su nombre se debe a que comprende un lenguaje formal que está diseñado para organizar algoritmos y procesos lógicos que serán luego llevados a cabo por un ordenador o sistema informático, permitiendo controlar así su comportamiento físico, lógico y su comunicación con el usuario humano, En la **figura 9** se muestra como ejemplo: C, C++, java, HTML, C#, PHP, XML, etc. cabe recalcar que todos tienen una misma idea programar, pero en diferentes plataformas. (Equipo editorial, 2021)

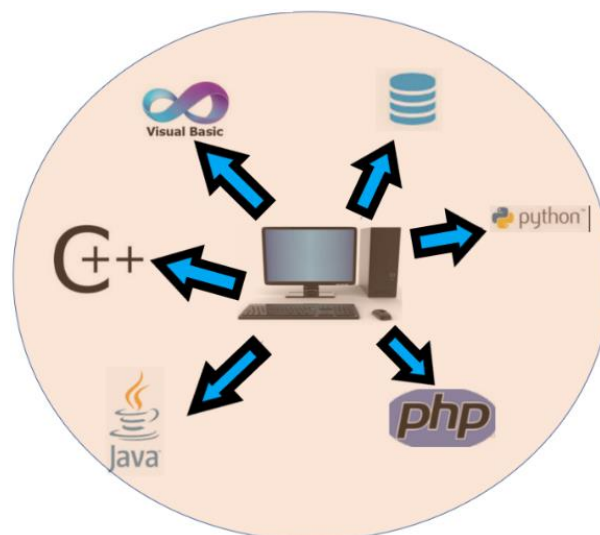
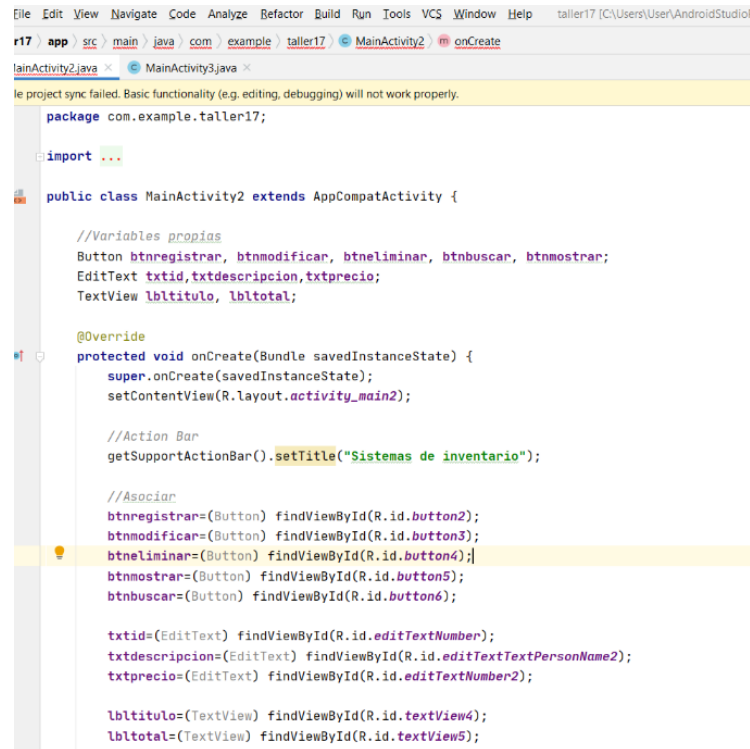


Figura 9: Lenguajes de programación

2.6.3.1 Java

En la **figura 10** se muestra Java es un lenguaje de programación y una plataforma informática que fue desarrollado por Sun Microsystems (ahora propiedad de Oracle

Corporation) en 1995. Se destaca por ser un lenguaje de programación orientado a objetos, lo que significa que se basa en la programación orientada a objetos (POO) para organizar y estructurar el código de manera eficiente (Content, 2019).



```
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help taller17 [C:\Users\User\AndroidStudioP
r17 > app > src > main > java > com > example > taller17 > MainActivity2 > onCreate
MainActivity2.java x MainActivity3.java x
The project sync failed. Basic functionality (e.g. editing, debugging) will not work properly.
package com.example.taller17;

import ...

public class MainActivity2 extends AppCompatActivity {

    //Variables propias
    Button btnregistrar, btnmodificar, btneliminar, btnbuscar, btnmostrar;
    EditText txtid,txtdescripcion,txtprecio;
    TextView lbltitulo, lbltotal;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main2);

        //Action Bar
        getSupportActionBar().setTitle("Sistemas de inventario");

        //Asociar
        btnregistrar=(Button) findViewById(R.id.button2);
        btnmodificar=(Button) findViewById(R.id.button3);
        btneliminar=(Button) findViewById(R.id.button4);
        btnmostrar=(Button) findViewById(R.id.button5);
        btnbuscar=(Button) findViewById(R.id.button6);

        txtid=(EditText) findViewById(R.id.editTextNumber);
        txtdescripcion=(EditText) findViewById(R.id.editTextTextPersonName2);
        txtprecio=(EditText) findViewById(R.id.editTextNumber2);

        lbltitulo=(TextView) findViewById(R.id.textView4);
        lbltotal=(TextView) findViewById(R.id.textView5);
    }
}
```

Figura 10: Ejemplo de Código Java

2.6.3.2 Php

En la **figura 11** se muestra código PHP (Hypertext Preprocessor) es un lenguaje de programación ampliamente utilizado para desarrollar aplicaciones web y crear páginas web dinámicas. Su principal objetivo es facilitar la conexión entre el servidor y la interfaz de usuario, permitiendo que las páginas web sean generadas de manera dinámica, lo que significa que el contenido puede variar según la interacción del usuario o los datos proporcionados por una base de datos (Zouza, 2020).

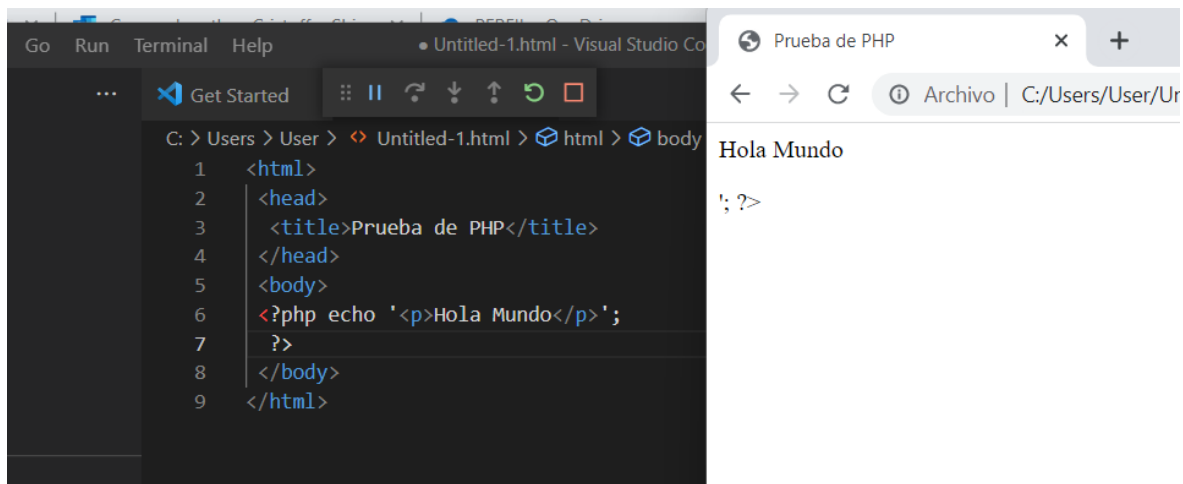


Figura 11: Ejemplo PHP "Hola Mundo"

2.6.3.3 HTML

En la **figura 12** se muestra código HTML (Hyper Text Markup Language) es un lenguaje de marcado utilizado para crear y dar formato a páginas web. Como bien mencionas, "Hyper Text" se refiere a los enlaces que permiten navegar a través de diferentes páginas y "Markup Language" significa que el lenguaje se utiliza para marcar o etiquetar el contenido de la página con elementos estructurales y de formato (Bravo, 2019).

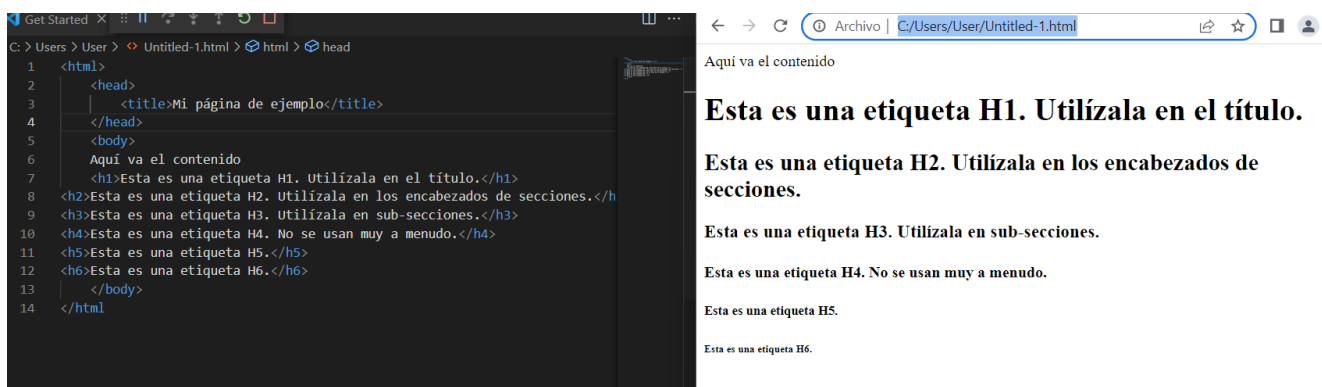


Figura 12: Uso de HTML

2.6.3.4 XML

En la **figura 13** se muestra código XML (Extensible Markup Language) es un lenguaje de marcado que se utiliza para codificar y estructurar datos en forma de texto legible tanto para las personas como para las computadoras. A diferencia de HTML, que está diseñado específicamente para la creación y visualización de páginas web, XML es un lenguaje genérico y versátil que se puede utilizar en una amplia variedad de aplicaciones (Zouza, 2020).

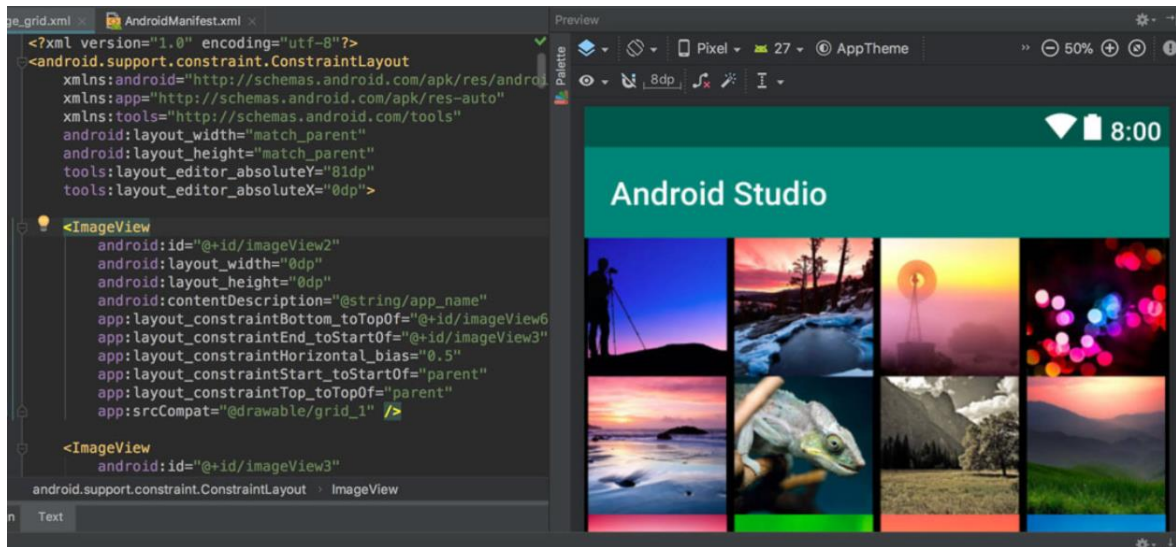


Figura 13: Código XML

2.7 Virtualización

La virtualización utiliza software para simular las características del hardware y generar un sistema informático virtual. Esto habilita a las organizaciones de tecnología de la información para ejecutar varios sistemas virtuales, incluyendo múltiples sistemas operativos y aplicaciones, en un único servidor (Vmware, 2022).

2.7.1 Plataformas de Virtualización

La virtualización de plataformas es la capacidad de ejecutar varios sistemas operativos de invitado no relacionados en un entorno incluido además de hardware compartido.

Tabla 2: Comparativa de máquinas virtuales y compatibilidad fog computing

Características	Compatibilidad con fog computing
VMware	
<ul style="list-style-type: none"> • Numerosas herramientas y funciones para entornos empresariales • Compartir archivos entre el host y el sistema virtualizado • Restaurar el estado de una VM fácilmente • Se integra con vSphere/ESXi y vCloud Air • Compatible con lectores de tarjetas inteligentes 	<p>Las redes virtuales ofrecen una serie de ventajas a las empresas: desde la reducción de la inversión en capital y los costes de mantenimiento hasta la segmentación sencilla de las redes. De esta</p>

<ul style="list-style-type: none"> • Soporte para USB 3.0 • Herramienta para compartir máquinas virtuales • Herramienta multi-plataforma • Herramienta para compartir máquinas virtuales • Cuenta con una herramienta para compartir máquinas virtuales 	<p>forma se pueden crear distintas conexiones de red para crear así áreas locales que son el fundamento principal de fog computing.</p>
VirtualBox	
<ul style="list-style-type: none"> • Controlar través de símbolo de sistema • Herramientas especiales para compartir archivos entre máquinas • Crear instantáneas para restaurar el estado anterior de una VM fácilmente • Soporte limitado para gráficos 3D • Herramienta para compartir máquinas virtuales • Herramienta multi-plataforma compatible con Windows, macOS, Linux y Solaris. • Herramienta para compartir máquinas virtuales. • Herramienta para compartir máquinas virtuales. • Herramienta multi-plataforma • Compatible con las máquinas virtuales de VMware. • Herramienta de captura de vídeo. • Cifrado de unidades virtuales • Soporte para puertos USB 2.0 y 3.0 	<p>Permite la creación de redes virtuales desde diferente tipo de conexiones de red entre las cuales se incluyen NAT, Red NAT, puente interno, Red interna, entre otros, estas permiten llegar a crear entorno como fog computing con la correcta configuración de adaptadores de red, sin mencionar que VirtualBox es compatible con VMware</p>

2.7.2 Sistema Operativo Windows Server 2016

Windows Server 2016, desarrollado por Microsoft, es un sistema operativo de servidor que ha ganado popularidad entre empresas, profesionales de TI y usuarios. En su creación, el equipo de Windows Server trabajó en estrecha colaboración con System Center y el equipo de Azure para crear un ecosistema compacto. Gracias a su cooperación con Apache en directorios y funciones, es posible generar un script eficiente para la creación de sitios web. Además, la colaboración con PhPMyAdmin facilita la gestión y creación de instalaciones para grandes volúmenes de datos. Para mejorar el acceso a sitios web en la red local a través

de redes celulares o externas, se ha utilizado un programa equivalente al símbolo del sistema llamado ngrok.

2.8 Base de Datos PhpMyAdmin

En la **figura 14** se muestra la interfaz de PhpMyAdmin, una aplicación web popular basada en PHP, se puede administrar bases de datos MySQL de manera sencilla y con una interfaz amigable. Una de las ventajas de utilizar esta aplicación web es que permite la conexión con servidores remotos, a los cuales no siempre se puede acceder mediante programas de interfaz gráfica. Esto facilita la gestión de bases de datos incluso en entornos donde el acceso directo a los servidores puede estar limitado. (Garcias, 2021)

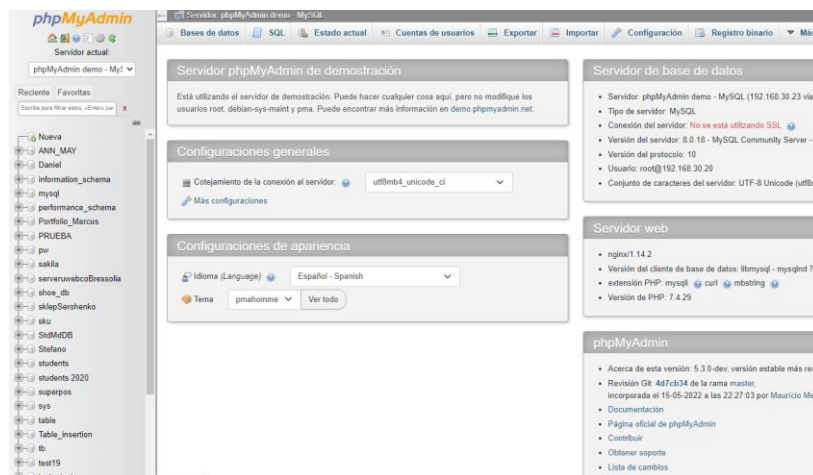


Figura 14: Interfaz PhpMyAdmin

2.9 Metodologías ágiles

Por definición, las metodologías ágiles son aquellas que permiten adaptar la forma de trabajo a las condiciones del proyecto, consiguiendo flexibilidad e inmediatez en la respuesta para amoldar el proyecto y su desarrollo a las circunstancias específicas del entorno, entre las metodologías más conocidas están: SCRUB, XP y KANBAN (Sotomayor, 2022)

2.9.1 Metodología XP



Figura 15: Estructura programación Xp

En la **figura 15** se puede apreciar las diferentes fases que contiene esta metodología en donde, Se planifica para un correspondiente diseño sea simple o prototipo, por consiguiente, se codifica y si es el caso se modifica para un mejor funcionamiento y por ende se prueba el producto para un lanzamiento óptimo después de todo el proceso se obtiene un software de calidad aplicando las normas ISO 25000.

2.10 Norma ISO 25000

Es un conjunto de normas establecidas por investigadores mediante pruebas científicamente aprobadas en donde se evalúa los criterios de adecuación funcional, eficiencia de desempeño, compatibilidad, usabilidad, fiabilidad, seguridad, mantenibilidad, portabilidad.

CALIDAD DEL PRODUCTO SOFTWARE							
Adecuación Funcional	Eficiencia de Desempeño	Compatibilidad	Usabilidad	Fiabilidad	Seguridad	Mantenibilidad	Portabilidad
<ul style="list-style-type: none"> • Completitud funcional • Corrección funcional • Pertinencia funcional 	<ul style="list-style-type: none"> • Comportamiento temporal • Utilización de recursos • Capacidad 	<ul style="list-style-type: none"> • Coexistencia • Interoperabilidad 	<ul style="list-style-type: none"> • Inteligibilidad • Aprendizaje • Operabilidad • Protección frente a errores de usuario • Estética • Accesibilidad 	<ul style="list-style-type: none"> • Madurez • Disponibilidad • Tolerancia a fallos • Capacidad de recuperación 	<ul style="list-style-type: none"> • Confidencialidad • Integridad • No repudio • Autenticidad • Responsabilidad 	<ul style="list-style-type: none"> • Modularidad • Reusabilidad • Analizabilidad • Capacidad de ser modificado • Capacidad de ser probado 	<ul style="list-style-type: none"> • Adaptabilidad • Facilidad de instalación • Capacidad de ser reemplazado

Figura 16: Normas de calidad del producto software

(Normas ISO 25000, 2023)

2.10.1 Eficiencia de desempeño (Rendimiento)

Evaluando los subcriterios de comportamiento temporal, utilización de recursos, capacidad se podrá determinar el rendimiento en diferentes casos gracias a las normas ISO 25000 como se muestra en la **figura 16**.

3. CAPÍTULO III. METODOLOGIA.

3.1 Metodología

El trabajo se caracteriza por ser una investigación bibliográfica, que utiliza una metodología mixta (cuantitativa y cualitativa) según el objeto de estudio. La metodología seleccionada para el estudio es la programación extrema (XP), la cual se basa en cuatro pilares fundamentales: planificación, diseño, codificación y pruebas. Además, se emplea un enfoque mixto para evaluar los criterios de rendimiento de la Aplicación Web a través del programa jmeter. Esta evaluación será considerada en la sección de pruebas del aplicativo web para la administración de medicamentos en pacientes ambulatorios.

3.2 Tipo y diseño de la investigación

3.2.1 Según la fuente de investigación

Es una investigación bibliográfica porque se realizó una búsqueda de información, accediendo a revistas, libros, base de datos científicas, tesis, entre otras fuentes referentes al tema de investigación, permitiendo comprender conceptos relacionados a fog computing en el ámbito de la salud para el desarrollo de un prototipo a escala mínima, permitiendo tener una visión más amplia del tema.

3.2.2 Según el objeto de estudio

Es una investigación de campo ya que se realizó un proceso de recolección y análisis de los requerimientos en la comunidad Sindy, además de un levantamiento de información sobre con el médico de la comunidad Sindy sobre la toma de medicamentos en pacientes ambulatorios.

3.2.3 Según el tipo de variable

Es una investigación cuantitativa-cuantitativa porque se analizó la percepción y valoración de los usuarios ante el aplicativo móvil de toma de medicamentos en pacientes ambulatorios y la participación del aplicativo web en la mejora de la toma de medicamentos en pacientes ambulatorios, específicamente en la parte de la programación y emisión automática de las alertas para pacientes.

3.3 Unidad de Análisis

La unidad de análisis fueron los pacientes ambulatorios de la comunidad Sindy provincia del Napo que durante el tiempo que se llevó a cabo este trabajo de investigación en el año 2022.

3.3.1 Población y muestra

Para el presente estudio la población y muestra será infinita, puesto que la investigación se lo hace mediante simulaciones jmeter para evaluar el rendimiento del aplicativo.

3.4 Entrevistas

Con el objetivo de recolectar información sobre la toma de medicamentos en pacientes ambulatorios y los requerimientos para la aplicación móvil y web que se ha realizado una entrevista al Médico de la comunidad Sindy, en el cual se ha usado una terna de preguntas para la entrevista, ver **Anexos Entrevista**.

3.5 Encuestas

Como sondeo inicial se realizó una breve encuesta a cada paciente con el objetivo de conocer sus criterios sobre la toma de medicamentos y la implementación de una aplicación móvil y web de toma de medicamentos en pacientes ambulatorios en Sindy, véase en los **Anexos Encuestas**.

3.6 Técnicas de análisis e interpretación de la información

Una vez obtenida la información de las entrevistas y encuestas se realizó un análisis para definir una orientación sobre el desarrollo de la aplicación web y móvil, teniendo en cuenta

ciertos criterios para el correcto funcionamiento de este en el que también se ha considerado algunas recomendaciones sobre la presentación de contenidos.

Como herramienta para la recolección de la información se ha utilizado Microsoft Word y Google Forms que es un software de administración de encuestas.

3.7 Desarrollo aplicativo web y móvil

Para el desarrollo de la aplicación web y móvil se ha empezado con la instalación y configuración de las herramientas informáticas, las cuales serán descritas a continuación y se evidencian en la **figura 17 se observa**: el aplicativo de desarrollo móvil Android Studio, el software de virtualización denominado Virtual Box y su respectiva máquina virtual con sistema operativo Windows server 2016 que tiene apache2.

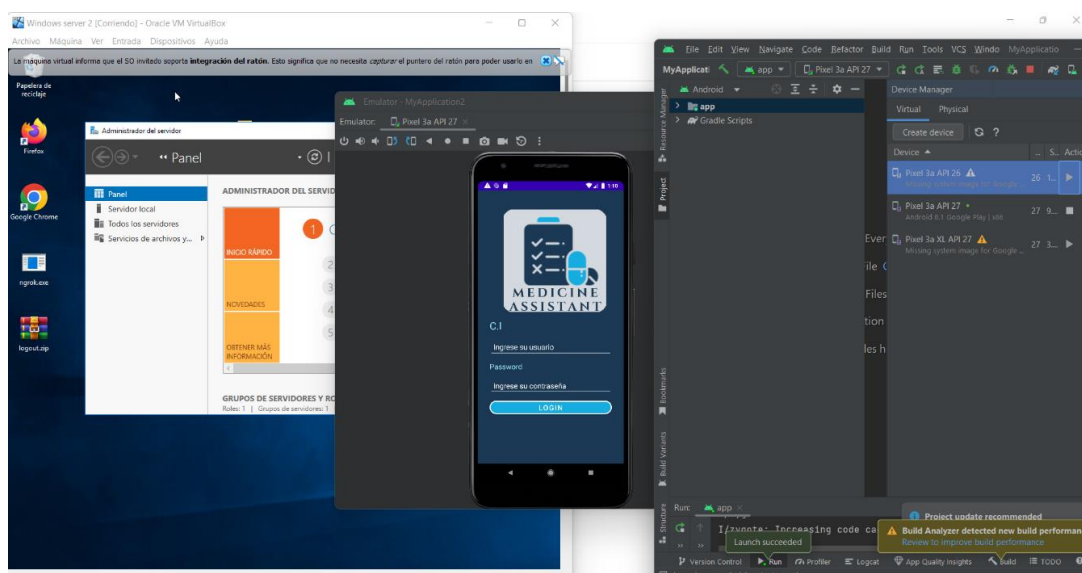


Figura 17: Herramientas de desarrollo web y móvil

En este trabajo el doctor Elder Grefa aportó atributos significativos que el aplicativo debe tener, además se tomó en cuenta a los pacientes ambulatorios de la comunidad Sindy sobre el uso de un nuevo aplicativo para el consumo de medicamentos además se usó la metodología XP para el desarrollo de la aplicación móvil por el hecho que posee 4 fases sumamente sencillas de usar, ya que la metodología XP se orientara para desarrollar la aplicación móvil de control y vigilancia en pacientes ambulatorios bajo la plataforma fog computing, a continuación, se describen las actividades realizadas en cada una de las fases:

3.7.1 Fase de Exploración

En esta etapa, se llevó a cabo un análisis de las necesidades de la comunidad Sindy mediante encuestas de opción múltiple dirigidas a los pacientes. Además, se realizó una entrevista con el doctor Elder Grefa, quien es médico de la comunidad. El doctor proporcionó información relevante sobre los datos esenciales que deben ser considerados en la aplicación web para el control y seguimiento de la toma de medicamentos. Asimismo, se valoró la perspectiva de los pacientes acerca de la implementación de una aplicación móvil para el seguimiento de la toma de medicamentos en pacientes ambulatorios de la comunidad Sindy, utilizando la plataforma de fog computing. También se evaluó la disponibilidad de sus dispositivos para utilizar esta aplicación y se determinaron los niveles de rendimiento recomendados para la misma.

The image shows a web-based survey titled "Encuesta de requisitos". It contains several sections of questions:

- Header:** "Encuesta de requisitos" with a link to "Iniciar sesión en Google para guardar lo que llevas hecho. Más información".
- Age Selection:** "Seleccione el rango de edad al que pertenece" with radio buttons for "15 a 18 años", "18 a 27 años", and "27 años o mas".
- Smart Device Usage:** "¿Cuenta con alguno de estos dispositivos inteligentes?" with checkboxes for "Celular", "Computador", and "Tablet".
- App Download Frequency:** "¿Con que frecuencia descarga aplicaciones en su móvil?" with radio buttons for "Una/Dos aplicaciones por día", "Una/Dos aplicaciones por semana", "Una/Dos aplicaciones por mes", and "No suele descargar aplicaciones".
- Internet Access:** "¿Cuenta con servicio de internet?" with radio buttons for "Si" and "No".
- Medication Schedule:** "¿Se considera una persona que toma sus medicamentos en las horas establecidas que le asigna el Doctor?" with radio buttons for "Si", "No", and "A veces".
- Social Circle:** "¿Conoce alguien en su círculo social que se le olvide tomar sus medicamentos?" with checkboxes for "Familia", "Amigos", and "Otro:" followed by a text input field.
- Mobile App Importance:** "¿Consideraría importante una aplicación móvil para hacer un recordatorio en la toma de sus medicamentos?" with radio buttons for "Si" and "No".

Figura 18: Encuesta a los pacientes de la comunidad Sindy

Entrevista

1. Qué necesidad Ud. prioriza en el sector de la salud?

Para el sector de la salud existen varias deficiencias que deben ser copadas de alguna manera, una de ellas es: como saber tomar los medicamentos, puesto que hay pacientes con graves enfermedades que no saben interpretar las indicaciones, además de eso los horarios muchas de las veces no se la entiende por motivos sea de lectura o escritura, para ello se necesitaría un aplicativo para la toma correcta de medicamentos que envíe notificaciones y que pueda monitorear si toma o no en los periodos asignados.

2. Cuál es la utilidad de aplicativos móviles y web en la salud?

La utilidad de estos aplicativos hace que la vida de las personas sea mucho más fácil y eficiente puesto que por medio de estos en los sectores de la salud, educación, industria y mas se ha innovado de manera exponencial, en el sector de la salud ayuda al monitoreo, interpretación y mas de resultados de exámenes y saber signos vitales de manera remota esto aporta a un sondeo en tiempo real de lo que se necesita.

3. Que características tendría el aplicativo que usaría para el sector de la salud?

El aplicativo que necesitaría para el monitoreo tendría una etapa de autenticación además tendría un banner con tres opciones en el cual se puede apreciar el registro y búsqueda de medicamentos, el registro y búsqueda de usuarios y la de receta para poder enviar a los pacientes en diferentes dosis, horarios y asignar a que paciente puede enviar de manera que sepa el tipo de medicamento que le toque y demás características.

4. Ud. considera importante la implementación de un aplicativo de notificaciones?

Considero importante la implementación de un aplicativo de notificaciones por la razón de que hay que estar a la vanguardia de las nuevas tecnologías y usarlas de manera que puedan ser de provecho para la respectiva notificación en tiempo real de la receta que debe tomar y el horario que debe tener.

5. Que impacto tendría un aplicativo para la toma de medicamentos?

Tendría un impacto positivo en la salud de los pacientes puesto que al cumplir con los medicamentos planteados, estos hacen posible que su salud no se deteriore o se controle de manera que será un aporte significativo para el sector de la salud e implementarse en otras áreas o sectores del sector social.

Figura 19: Resultado de entrevista a Doctor Elder

Debido a esto las encuestas se lo ha realizado durante dos semanas y los resultados se muestran en la sección correspondiente de resultados. Posteriormente se ha procedido a la digitalización de los materiales y contenidos proporcionados en las reuniones, y a partir de los cuales se ha elaborado los textos escritos que van a mostrarse dentro de la aplicación, procurando una redacción clara y estructurada.

3.7.2 Fase de Planificación

Tabla 3: Estructura de la planificación

Estructura de la planificación			
1. Plan de entregas	En este segmento de la estructura de planificación se indicará la duración que se tomará en cuenta para el desarrollo de cada complemento	Diseño	1 Semanas
		Desarrollo de diseño	2 Semanas
		Implementación de diseño	1 Semana
		Programación	3 Semanas
		Ejecución	1 Semana
		Implementación del aplicativo	1 Semana

Plan de Iteraciones	En este segmento de la estructura de planificación se indicará los procesos de iteración que se presentaran en un tiempo determinado de días.	Diseño	
		<ul style="list-style-type: none"> - Reuniones con los involucrados del proyecto. - Borrador del diseño - Reunión para aprobación - Corrección del diseño - Aprobación del Diseño Final 	
		Desarrollo de diseño	
		<ul style="list-style-type: none"> - Reuniones con los involucrados del proyecto. - Borrador del diseño - Reunión para aprobación - Corrección del diseño - Aprobación del Diseño Final 	
		Implementación de diseño	
		<ul style="list-style-type: none"> - Reuniones con los involucrados del proyecto. - Borrador del diseño - Reunión para aprobación - Corrección del diseño - Aprobación del Diseño Final 	
Reuniones con el tutor	En este segmento de la estructura de planificación se indicará los días y la hora de reuniones	Programación	
		<ul style="list-style-type: none"> - Reuniones con los involucrados del proyecto. - Borrador del aplicativo - Reunión para aprobación - Corrección del aplicativo - Aprobación del aplicativo 	
		Ejecución	
		<ul style="list-style-type: none"> - Reuniones con los involucrados del proyecto. - Borrador del aplicativo - Reunión para aprobación - Corrección del aplicativo - Aprobación del aplicativo Final 	
		Implementación del aplicativo	
		<ul style="list-style-type: none"> - Reuniones con los involucrados del proyecto. - Borrador del aplicativo - Reunión para aprobación - Corrección del aplicativo - Aprobación del aplicativo final 	
		Diseño	Sábado 9pm
		Desarrollo de diseño	
		Implementación de diseño	
		Programación	
		Ejecución	

	con el tutor mediante Microsoft Teams	Implementación del aplicativo	
--	---	-------------------------------	--

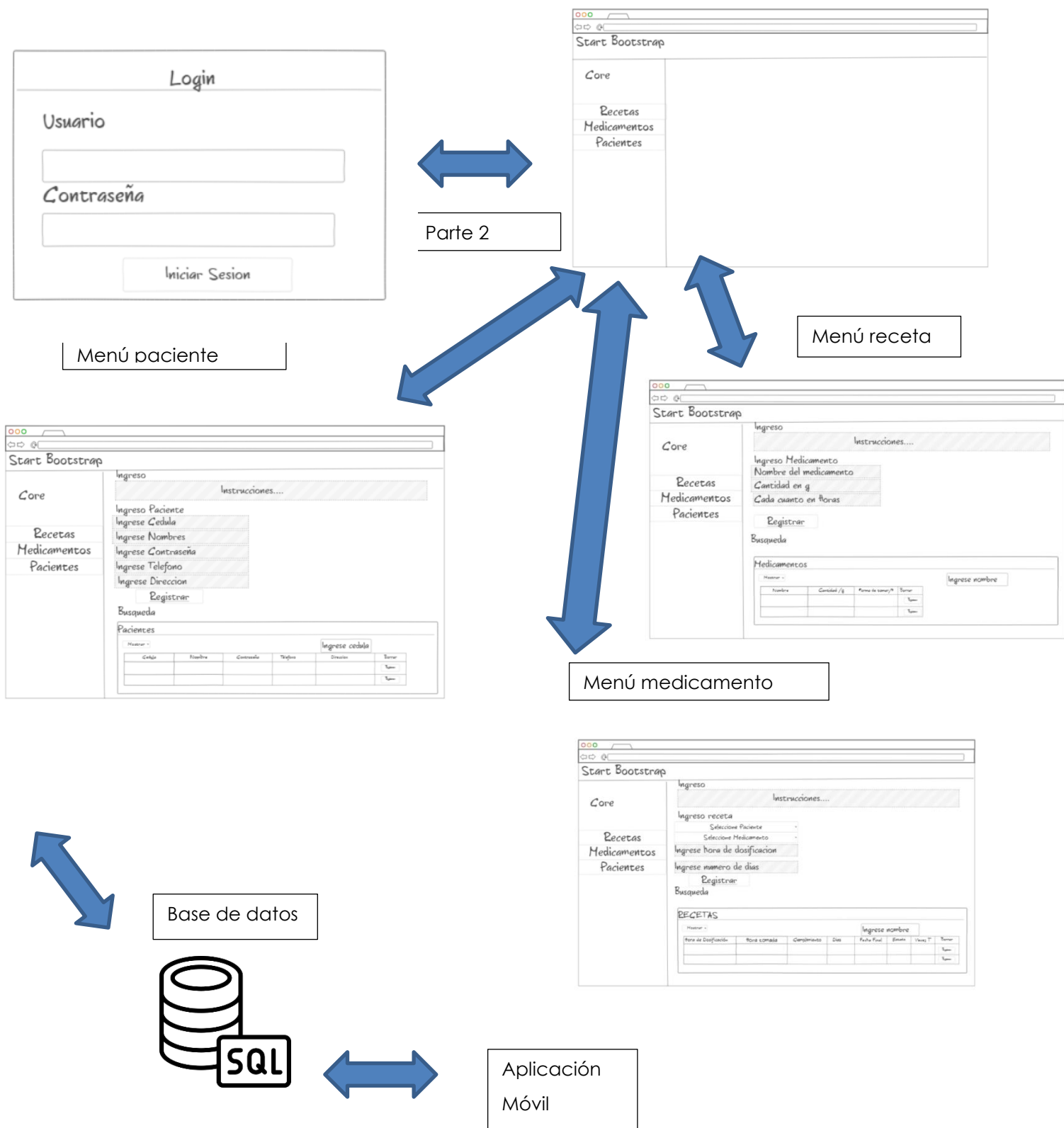
3.7.3 Diseño del aplicativo web

Parte 0

Login

Parte 1

Menú principal



Parte 0: En esta sección se creó un panel de autenticación o logueado, en el cual solo usuarios creados por el administrador del base de datos local pueden ingresar a cumplir las

diferentes tareas que necesite, cabe recalcar que si un usuario no se encuentra en el sistema de base de datos, este no podrá ingresar y se lo enviara a una página de error, todo esto se lo hace con el fin que la integridad del sistema web no sea vulnerado y los usuarios queden satisfechos con el servicio.

Parte 1: Se realizo una interfaz sencilla pero que cumplió con el objetivo de esta investigación en el cual se ha incorporado un módulo de recetas, medicamentos y paciente. Los cuales están acorde a los requerimientos del doctor los cuales se detallan en el siguiente apartado.

Parte 2: Se puede apreciar los diferentes módulos como:

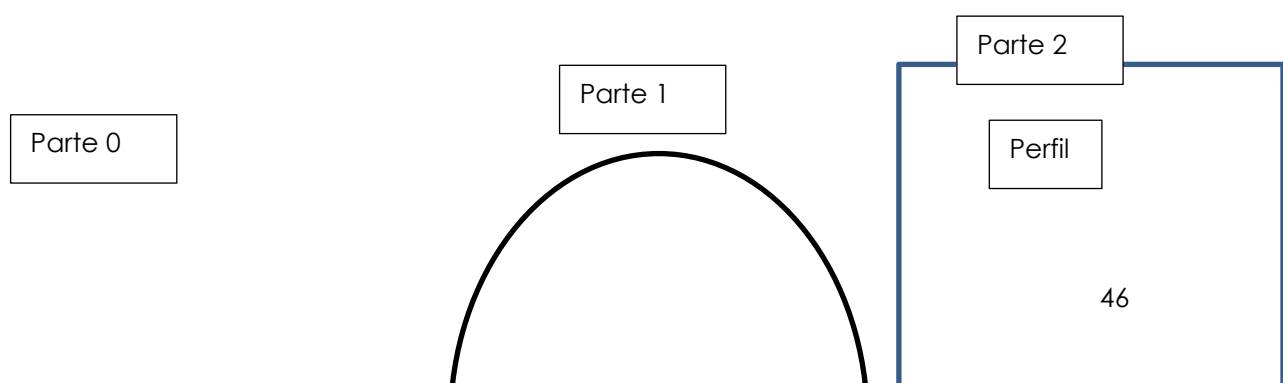
Medicamentos: En esta sección el doctor podrá añadir atributos del medicamento tales son nombre, cantidad en gramos y cada que tiempo debe hacerlo.

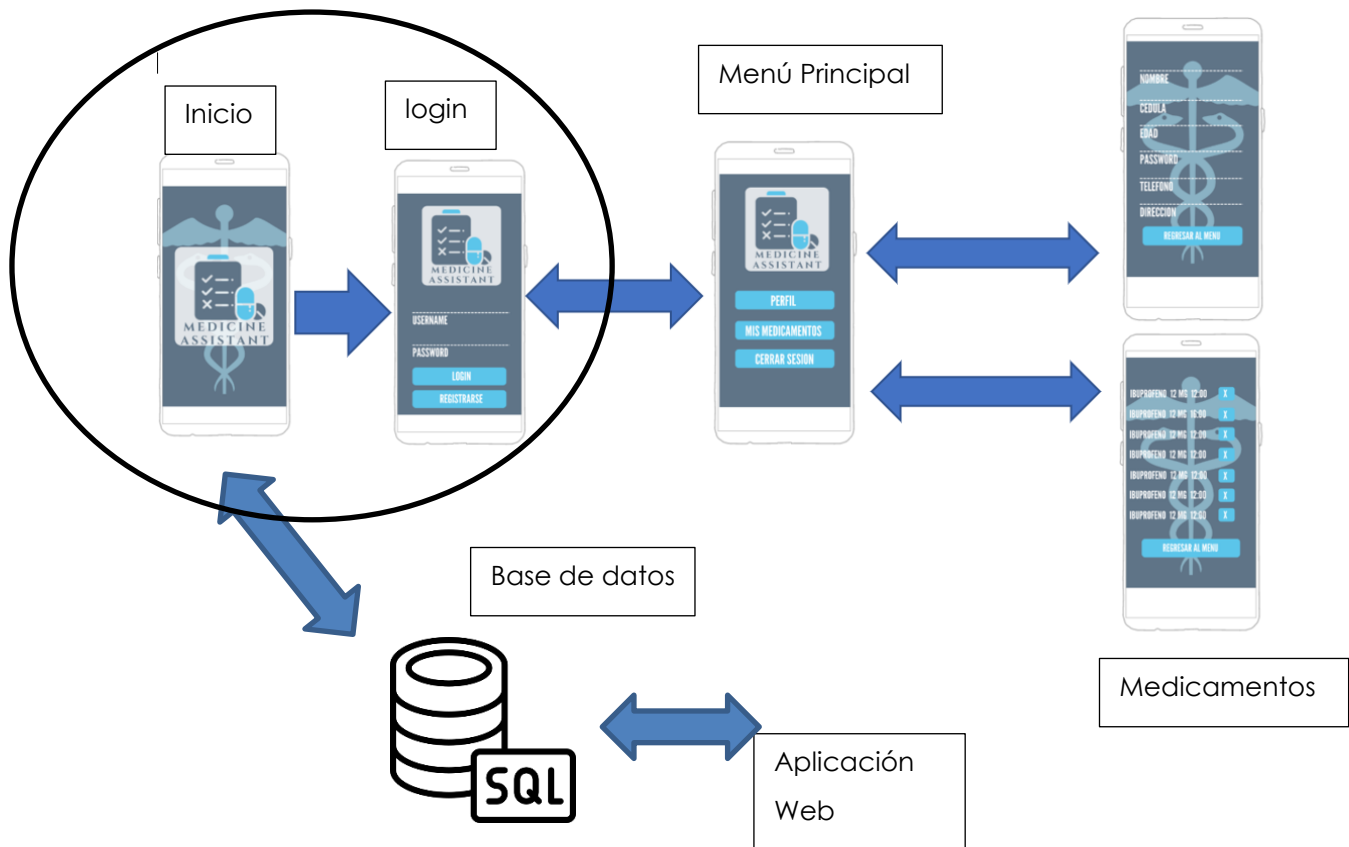
Paciente: En este apartado el doctor podrá añadir atributos como nombres, apellidos, dirección, teléfono, cedula y contraseña.

Recetas: Es la sección más importante para el doctor en esta podrá añadir los pacientes y medicamentos acorde al horario que considere para que se cree un horario de alertas que están previamente analizadas para el cumplimiento como el número de dosis que debe tomar y el cumplimiento que este debe tener para que no recaiga en otras enfermedades.

Todo este sistema web este entrelazado al sistema móvil y a su vez a la base de datos local en phpMyadmin que está a cargo del administrador y en ella se puede realizar los cambios que el doctor le sugiera a futuro.

3.7.4 Diseño del aplicativo móvil





Parte 0: Esta parte está compuesta de la interfaz de Splash e inicio de sesión, en el cual el usuario deberá ingresar un correo y contraseña previamente registrado.

Parte 1: En esta parte del diseño menú principal consta de la interfaz de 3 botones los cuales son: perfil y mis medicamentos. En el botón perfil es donde el usuario deberá tiene los siguientes campos: nombre, cedula, edad, teléfono, en el botón mis medicamentos se visualiza la información en el cual deberá tomar a la hora y cantidad que le asigne el Doctor, seguido de un botón llamado regresar al menú.

Parte 2: En esta parte del diseño el botón perfil es donde el usuario deberá tiene los siguientes campos: nombre, cedula, edad, teléfono, en el botón mis medicamentos se visualiza la información en el cual deberá tomar a la hora y cantidad que le asigne el Doctor, seguido de un botón llamado regresar al menú.

3.7.5 Diseño arquitectónico

Una vez que se determina el diseño de la interfaz de la aplicación web y móvil, se implementa los modelos arquitectónicos en el desarrollo de proyectos. A vez determinando los diseños de interfaces de la aplicación web y móvil, se establece el patrón arquitectónico que se ha implementado en el desarrollo del proyecto.

3.7.5.1 Patrón arquitectónico

Se ha empleado el patrón de arquitectura Modelo-Vista-Controlador (MVC) para dividir la lógica del sistema en tres capas distintas. El propósito de esta estrategia es facilitar la detección rápida de errores y permitir la reutilización efectiva de código tanto en la elaboración de la Aplicación Web como en la Móvil.

3.7.5.2 Sistema web

La **figura 20** representa la implementación del patrón arquitectónico para el Sistema Web, teniendo en cuenta las herramientas de desarrollo empleadas. Este enfoque permite que las funciones de cada capa trabajen de manera independiente y asegura una distribución adecuada del código para una eficiente separación de responsabilidades.

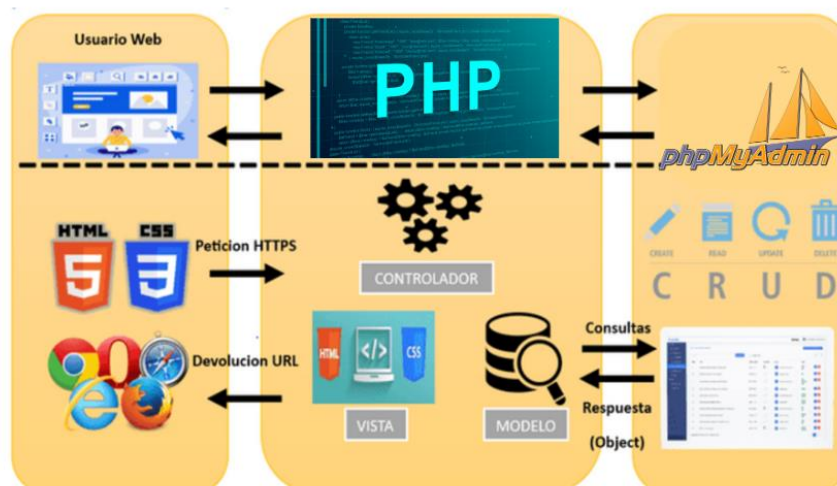


Figura 20: Arquitectura sistema web

3.7.5.3 Aplicación móvil

La **figura 21** muestra el patrón arquitectónico aplicado en la Aplicación Móvil, considerando las herramientas de desarrollo empleadas, en caso de que se quieran realizar actualizaciones o crear nuevas versiones en el futuro.

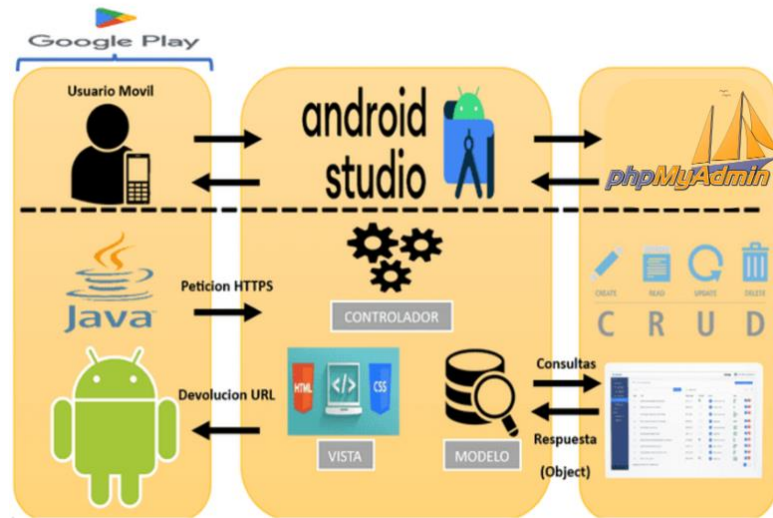


Figura 21: Arquitectura aplicativo móvil

3.7.6 Diseño de la infraestructura de fog computing

Para poder desarrollar el diseño de una infraestructura de fog computing, primero se debe de entender el principio del significado de fog computing, el cual implica la creación de una red de área local con un borde superior que se conecta a la red de Cloud Computing para poder permitir acceso a la red, mientras que en el borde inferior se encontrara los dispositivos (IoT) que permitirán la participación de los usuarios

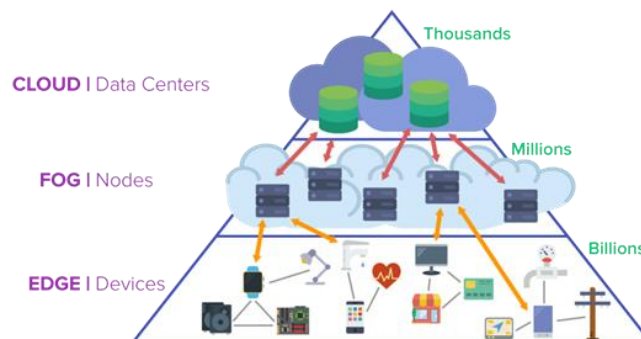


Figura 22: Arquitectura fog computing

Para el desarrollo de este entorno de red se realizó una conexión mediante una RED NAT, la cual se establece gracias a las herramientas de red que nos proporciona la plataforma de VirtualBox y a partir de una breve configuración de las tarjetas o adaptadores de red en cada máquina virtual se establece el tipo de conexión que tendrán correspondiente a la infraestructura de red Aplicada.

3.7.6.1 Fog computing

En la **figura 23** muestra el patrón arquitectónico de fog computing integrado con la Aplicación Web y Móvil.

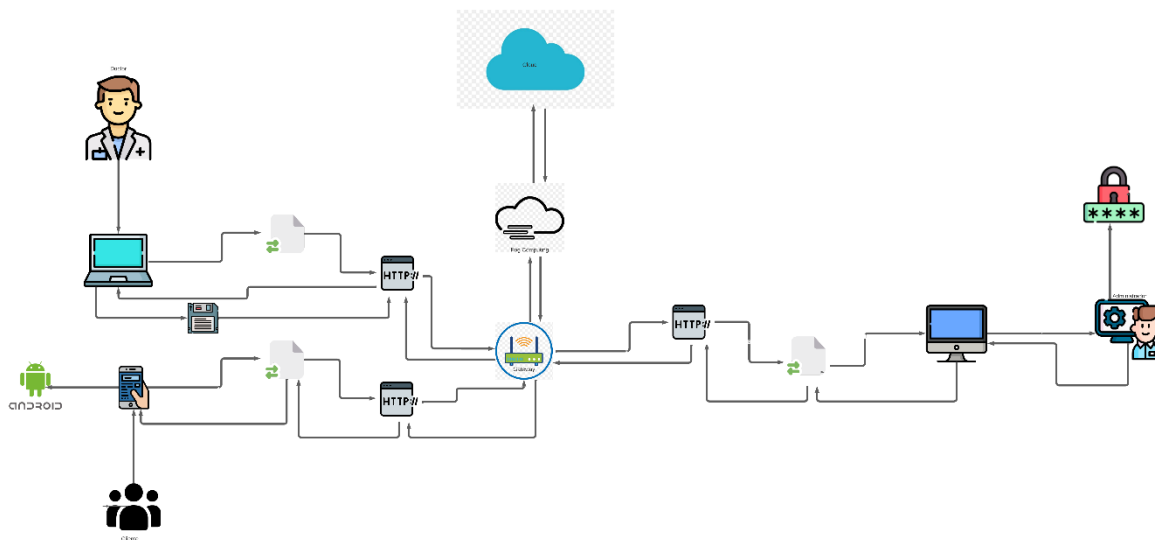


Figura 23: Arquitectura de fog computing integrada

3.8 Desarrollo

Se tendrá en cuenta 3 interfaces: Interfaz de paciente, medicamentos y receta.

En las cuales el medico puede ingresar recetas dependiendo el estado del paciente, así como todos los medicamentos que se requiera y finalmente la parte de agregar sus pacientes.

3.8.1 Desarrollo del aplicativo web

Para el desarrollo del aplicativo Web se realizaron diversos pasos (los mismos que se enlistan a continuación) los cuales se enlistarán a continuación con la correspondiente figura de aplicación:

- En la **figura 24** se muestra Medicine Assistant en el cual se creó los HTML junto a los CSS que serán el complemento para la ejecución del aplicativo web, dentro los cuales se involucra el login, receta, medicamento y pacientes

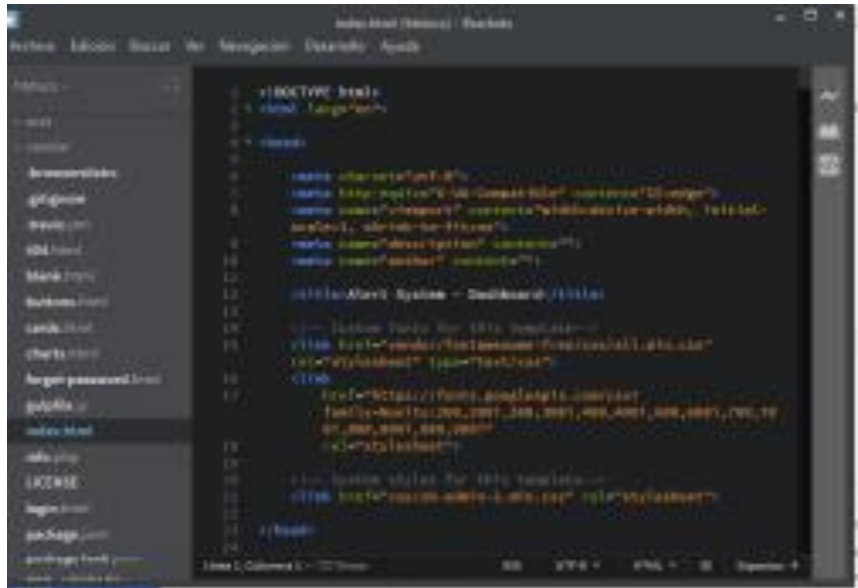


Figura 24: Código HTML y CSS en Visual Studio Code

3.8.1.1 Desarrollo de módulos:

- **Modulo Recetas**

En la **figura 25** se muestra los principales campos que deben ser llenados los cuales son; paciente, medicamento, Hora y número de días, que se realizan por medio de la aplicación Visual Studio Code en el cual se puede digitar en código HTML y PHP que hacen que conecte con la base de datos, los mismo que se detallan en la siguiente tabla:



Figura 25: Modulo Receta

En la **tabla 4** se muestra detalladamente la información de cada variable a usar en este módulo.

Atributo	Descripción
Paciente	Se ingresa el paciente al cual se le va a recetar el medicamento.
Medicamento	Se ingresa el tipo de medicamento que deberá tomar el paciente
Hora	Se ingresa el horario establecido para la toma del medicamento recetado en el que tomara el medicamento el paciente
Número de días	Se ingresa el número de días que dura el tratamiento, que debe tomar la receta

Tabla 4: Descripción de variables modulo Recetas

3.8.1.2 Modulo Medicamentos

En la **figura 26** se muestra los principales campos que deben ser llenados los cuales son; medicamento, cantidad y hora, los mismo que se detallan en la siguiente tabla:

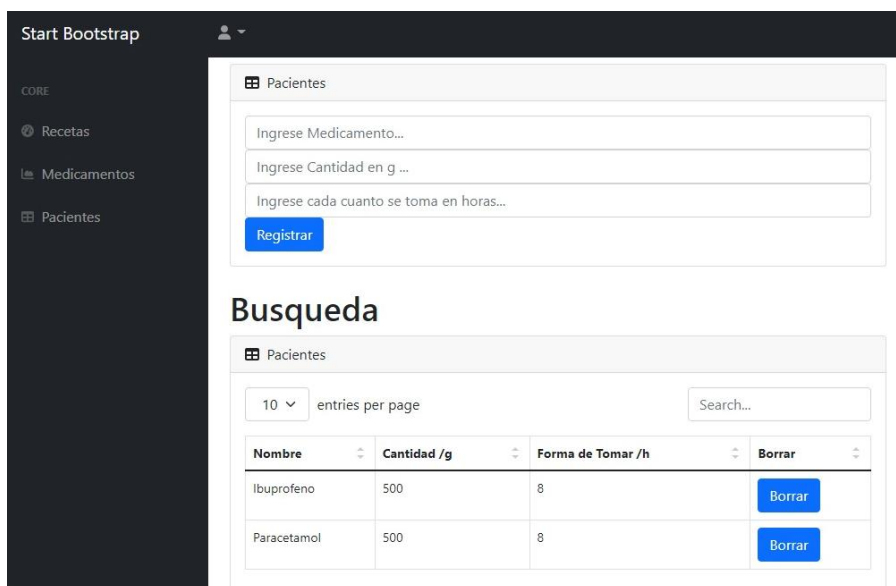


Figura 26: Modulo Medicamentos

En la **tabla 5** se detalla el uso de las variables.

Atributo	Descripción
Medicamento	Se ingresa el tipo de medicamento que deberá tomar el paciente.

Cantidad(gramos)	Se ingresa el horario establecido para la toma del medicamento recetado en el que tomara el medicamento el paciente.
Hora	Se ingresa la hora que deberá tomar el paciente el medicamento.

Tabla 5: Descripción de variables Medicamento

3.8.1.3 Modulo Pacientes

En la **figura 27** se muestra los principales campos que deben ser llenados los cuales son; cedula, nombres, contraseña, teléfono y dirección, los mismo que se detallan en la siguiente tabla:

Figura 27: Modulo Pacientes

En la **tabla 6** se muestra de forma detallada las diferentes variables usadas.

Atributo	Descripción
Cedula	Se ingresa el número de cédula del paciente.
Nombres	Se ingresa los nombres del paciente a tratar.
Contraseña	Se ingresa la contraseña del paciente.
Teléfono	Se ingresa el teléfono del paciente.
Dirección	Se ingresa la dirección del paciente.

Tabla 6: Descripción de variables de Pacientes

3.8.2 Base de datos

Para la realización de la base de datos se implementó una base de datos phpmyadmin para guardar y tener un registro de cada usuario al momento de usar el aplicativo web y móvil facilitando el envío de información referente a cada paciente, para esto se tomó en cuenta las siguientes entidades: Medico, Paciente, Medicamentos y Receta donde cada tabla estará estructurada con atributos que cumple con las necesidades y se adecuan a los tipos de datos requeridos para cada calculo necesario, a continuación se explicara la función de cada entidad y atributo: Entidad: Médicos Será el personal que reciba las notificaciones en el aplicativo WEB generadas por el aplicativo móvil cuando un paciente deba recibir las notificaciones en la hora de la toma de medicamento

Entidad: Médico

Será el personal que envíe las notificaciones mediante el aplicativo WEB generado en el aplicativo móvil las mismas para la toma oportuna de medicamentos, para esto se tomará en cuenta los siguientes atributos con sus funcionalidades:

Atributo	Variable	Descripción
Cedula	Varchar(10)	Se guardará la cedula del médico y se utilizará como clave primaria
Nombres	Varchar(200)	Se guardará los nombres completos del medico
Apellidos	Varchar(100)	Se guardará los apellidos completos del medico
Password	Varchar(60)	Se guardará la contraseña del medico
Dirección	Varchar(60)	Se guardará la dirección del medico
Celular	Varchar(60)	Se guardará el número de celular del medico

Tabla 7: Estructura en PHPmyadmin de entidad Medico

Entidad: Pacientes

Esta entidad es la que va a recibir las notificaciones en la hora de toma de sus medicamentos.

Atributo	Variable	Descripción
Cédula	Varchar(10)	Se guardará la cedula del paciente y se utilizará como clave primaria
Nombres	Varchar(150)	Se guardará los nombres completos del paciente
Apellidos	Varchar(150)	Se guardara los apellidos completos del paciente
Password	Varchar(150)	Se guardará la contraseña del paciente
Teléfono	Varchar(10)	Se guardará el teléfono del paciente
Dirección	Varchar(400)	Se guardará la dirección del paciente

Tabla 8: Estructura en PHPmyadmin de entidad Paciente

Entidad: Medicamentos

Esta entidad es la que guarda los medicamentos.

Atributo	Variable	Descripción
Id_m	Int(11)	Se guardará la identificación del medicamento y se utilizará como clave primaria
Nombre	Varchar(150)	Se guardará el nombre del medicamento
Cant	Varchar(10)	Se guardará la cantidad del medicamento
Tiempo	Int(3)	Se guardará la hora de la toma del medicamento

Tabla 9: Estructura en PHPmyadmin de entidad Medicamentos

Entidad: Receta

Esta entidad es la que guarda la receta.

Atributo	Variable	Descripción
Id_r	Int(11)	Se guardará la identificación de la receta y se utilizará como clave primaria
Cedula_p	Varchar(10)	Se guardará la cedula del paciente
Id_m	Int(11)	Se guardará la identificación del medicamento
horaR	Varchar(300)	Se guardará la hora de la receta
horaT	Varchar(300)	Se guardará la hora en la que tiene que tomar el medicamento el paciente
Porcentaje	Varchar(3)	Se guardará el porcentaje en la toma del medicamento
Ndias	Varchar(2)	Se guardará los días en los que tiene que tomar el medicamento
D_inicio	Varchar(150)	Se guardará la hora y el día de inicio
Estado	Varchar(100)	Se guardará el estado del paciente
Vecest	Int(11)	Se guardará el número de veces que toma el medicamento
Cedula_m	Varchar(10)	Se guardará la cedula del medico

Tabla 10: Estructura en PHPmyadmin de entidad Receta

En la **figura 28** se muestra las diferentes relaciones que se han creado con los diferentes tipos de variable que se han utilizado para el proceso de creacion de la aplicaci3n web.

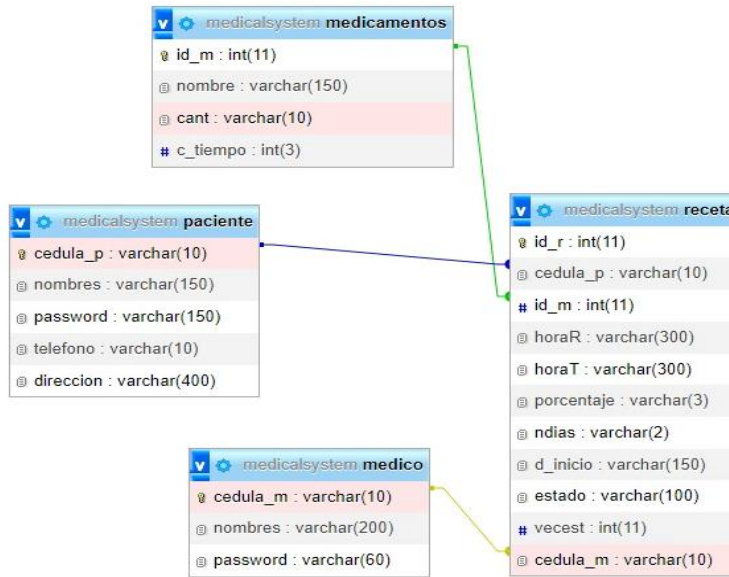


Figura 28: Diagrama E_R de la base de datos

3.8.3 Desarrollo del aplicativo m3vil

Para el desarrollo del aplicativo m3vil se realizaron diversos pasos los cuales se enlistar3n a continuaci3n con las im3genes propias del autor de layouts para las distintas interfaces que se requieren en el aplicativo m3vil y se configuro sus funciones, todo esto en el aplicativo Android studio con lenguaje java y XML

En la **figura 29** se muestra el desarrollo del splash de la pantalla principal del aplicativo se us3 c3digo libre con la ayuda de un delay para el retardo en la pantalla.

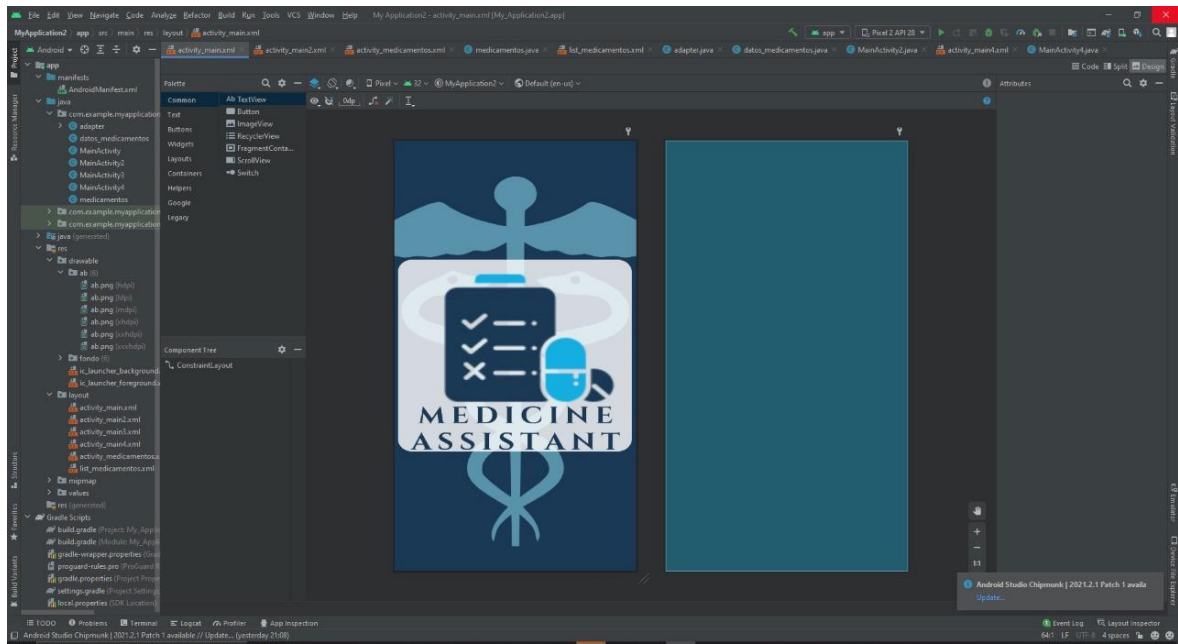


Figura 29: Interfaz de desarrollo de splash

En la **figura 30** se muestra el layout se tiene la pantalla de autenticación donde solo se podrá ingresar con los usuarios y contraseñas asignadas por el doctor a sus pacientes.

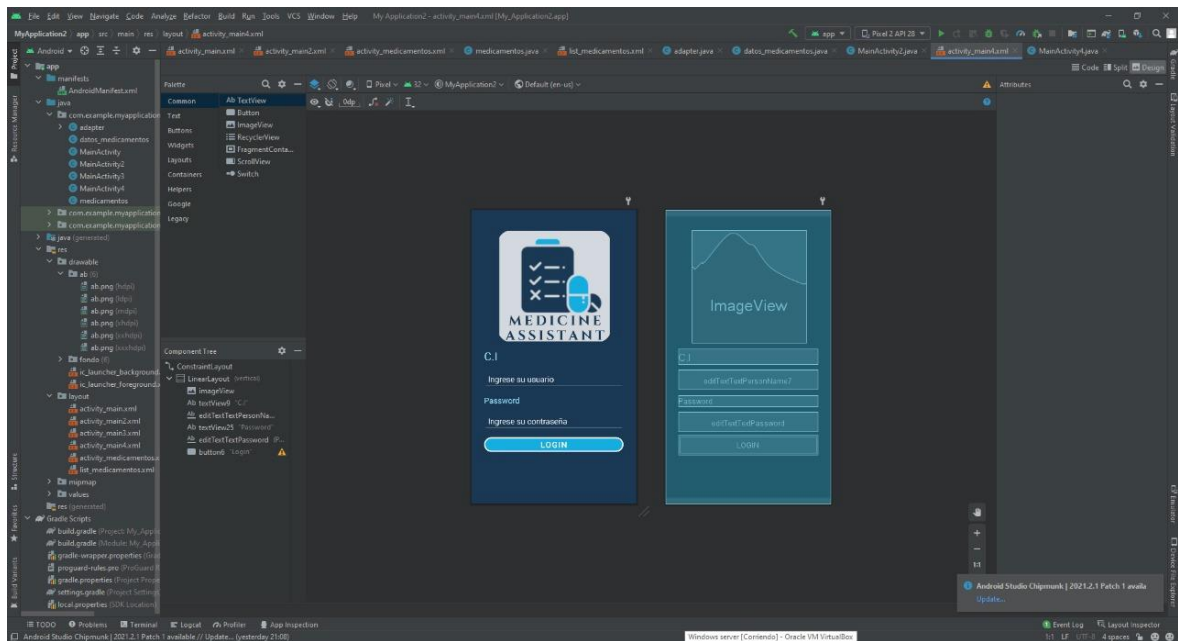


Figura 30: Interfaz de desarrollo de login

En la **figura 31** se muestra el layout se tiene el menú principal del paciente en donde constara el perfil, medicamentos y el cerrar sesión.

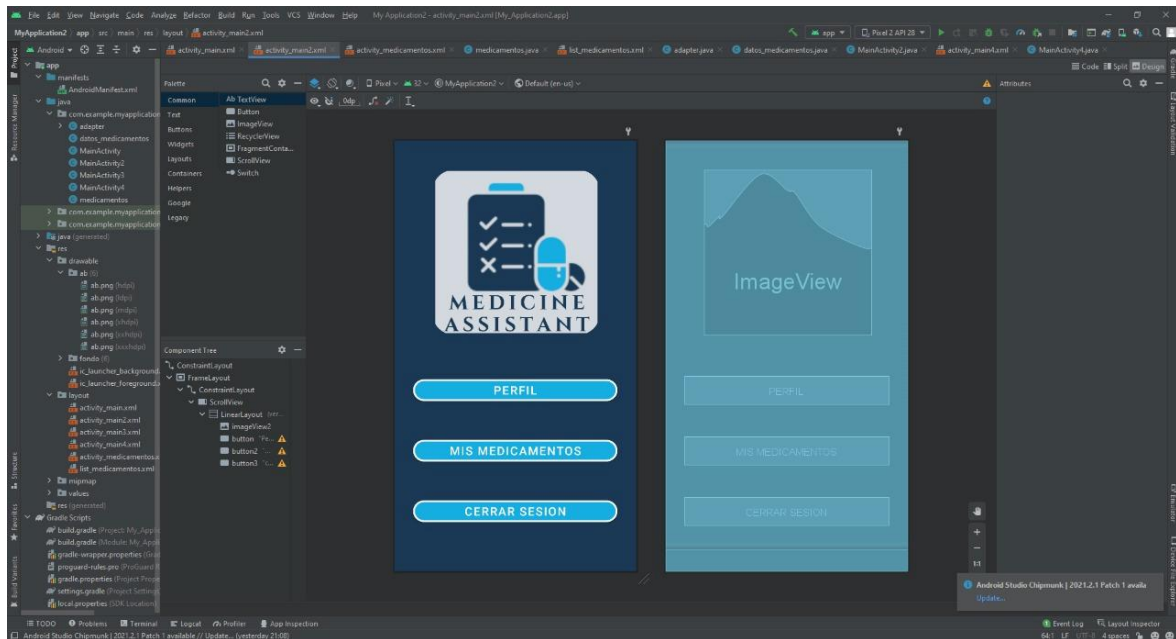


Figura 31: Interfaz de desarrollo de menú principal

En la **figura 32** se muestra el layout se tiene el perfil en donde se puede modificar la información relevante del paciente.

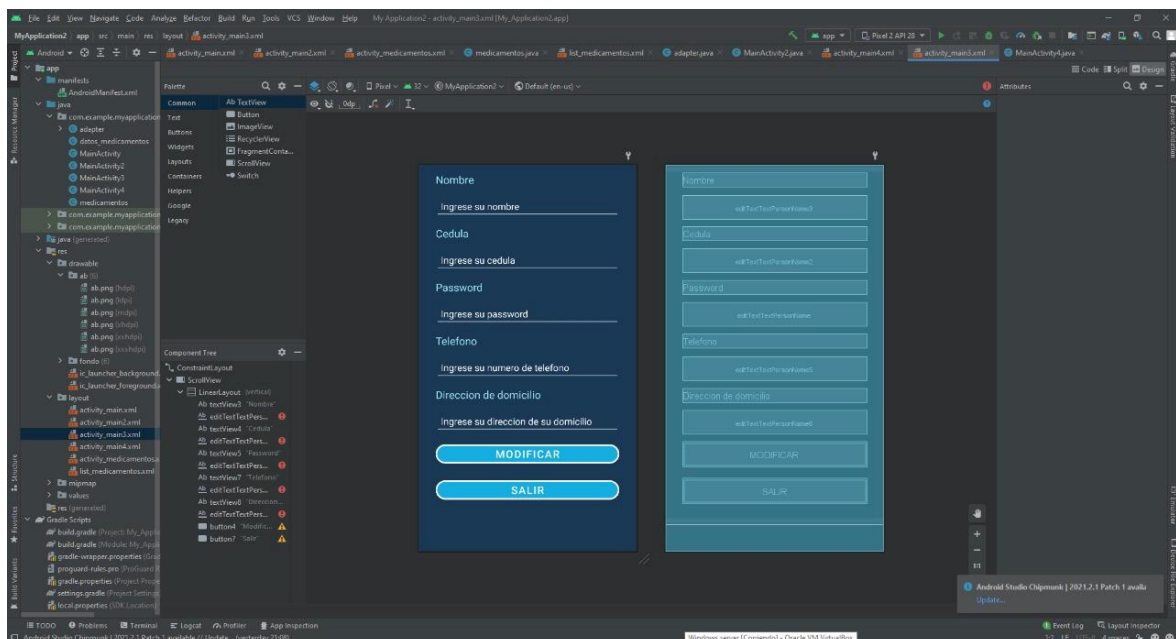


Figura 32: Interfaz de desarrollo de perfil de paciente

En la **figura 33** se muestra el más importante de los layout se tendrá el banner de medicamentos para aceptar la toma de medicamentos en los horarios establecidos.

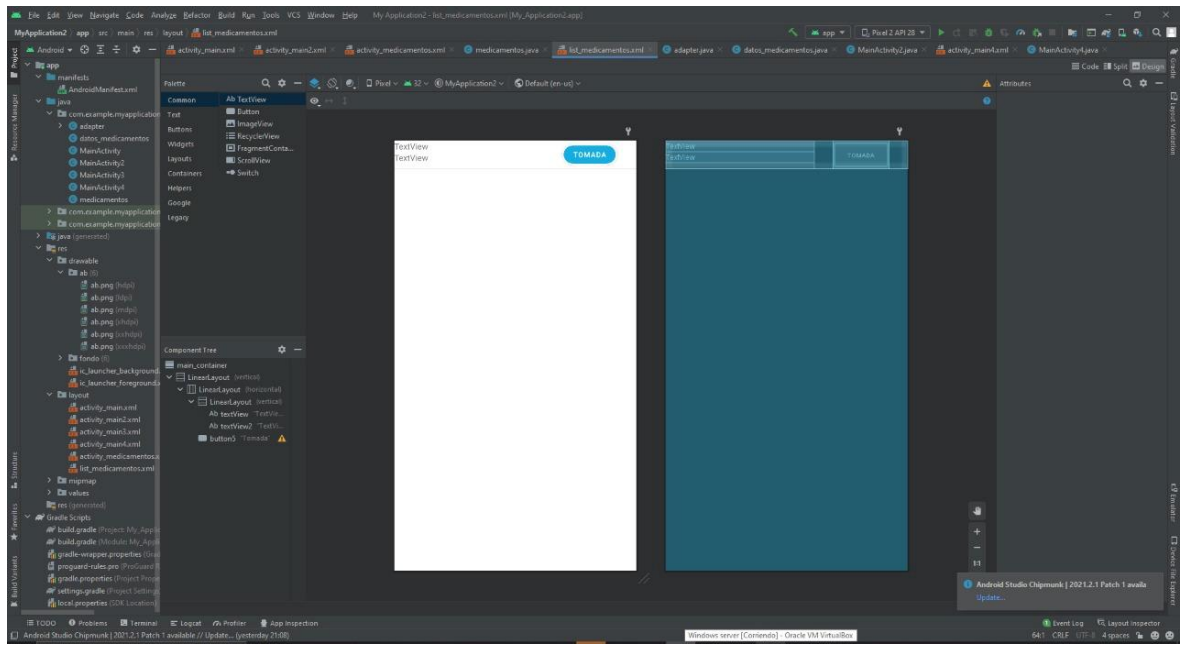


Figura 33: Interfaz de desarrollo de medicamentos

4. CAPÍTULO IV. RESULTADOS Y DISCUSIÓN

Para mostrar el rendimiento del sistema de vigilancia en el aplicativo web y móvil se ejecutará diversas pruebas mediante aplicativos Aida-64, Símbolo de sistema y jmeter para lograr ver los subcriterios de capacidad, utilización de recurso y comportamiento temporal, mismos que se reflejan al demostrar el correcto funcionamiento.

4.1 Resultado de pruebas

4.1.1 Pruebas unitarias

En este tipo de pruebas se comprobó la conectividad en el caso del fog computing en su entorno, además mediante el URL se ingresó por medio de diferentes tipos de aparatos como smartphones de gama media en adelante todo ira dependiendo de las actualizaciones que Android posea para establecer compatibilidad como por ejemplo el dispositivo a utilizar fue un Samsung Galaxy a32 de procesador MediaTek Dimensity 720,4 gb de ram y sistema Android versión 13 con 128b gb de memoria rom y computadoras con sistema operativo Windows en el caso del ejemplo que se ocupo fue una Hp Core i5-10350h con 16 gb de ram y 1 tb de almacenamiento con disco duro y 256 gb de almacenamiento para disco solido m.2 como se muestra en la **figura 34** y **figura 35**.

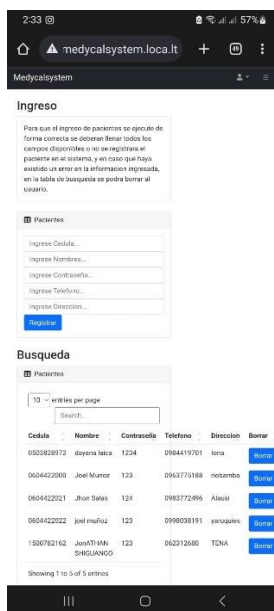


Figura 34: Conectividad con smartphone

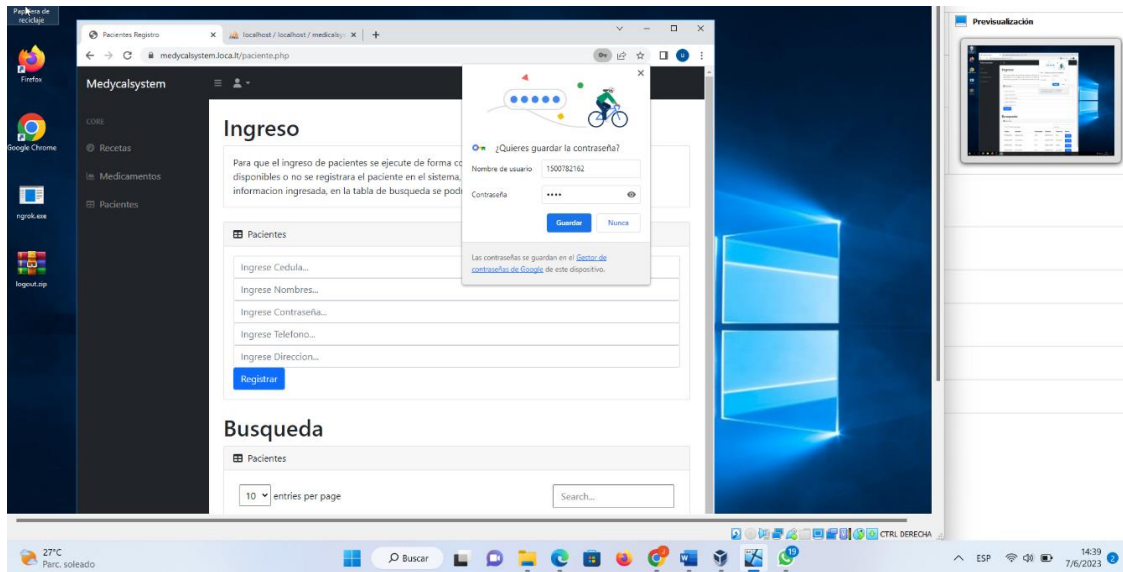


Figura 35: Conectividad con computadora

Para la interfaz web se pudo constatar que la información referente a la toma de medicamentos es la misma que está en la base de datos como se muestra en la **figura 36**.

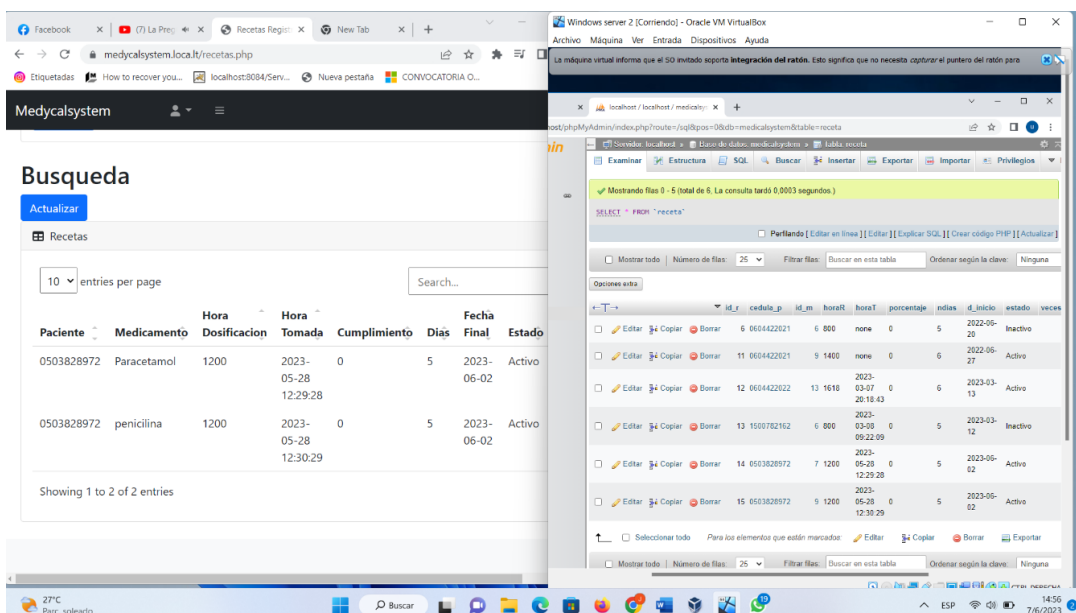


Figura 36: Verificación de información

Para el aplicativo móvil se verifico que toda la información que contenga el aplicativo móvil sea la misma que la base de datos refleja para evitar inconvenientes en procesamiento de datos como se muestra en la **figura 37**.

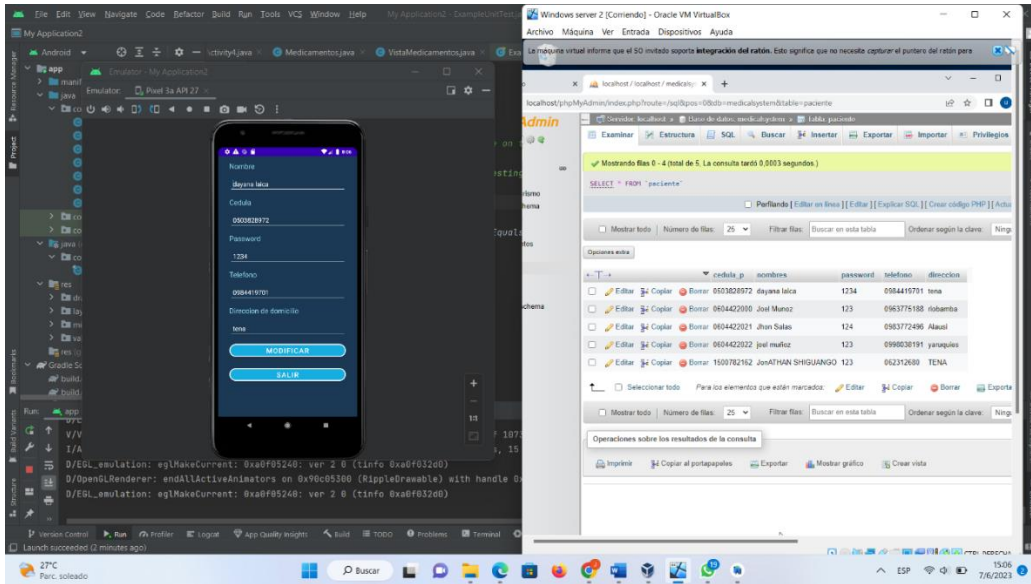


Figura 37: Verificación de información del paciente

4.1.2 Pruebas de aceptación

En las pruebas de aceptación o también denominada funcionales el paciente usará el aplicativo para demostrar que todo esté funcionando correctamente esto se basa en las anteriores historias de usuario que son una serie de iteraciones.

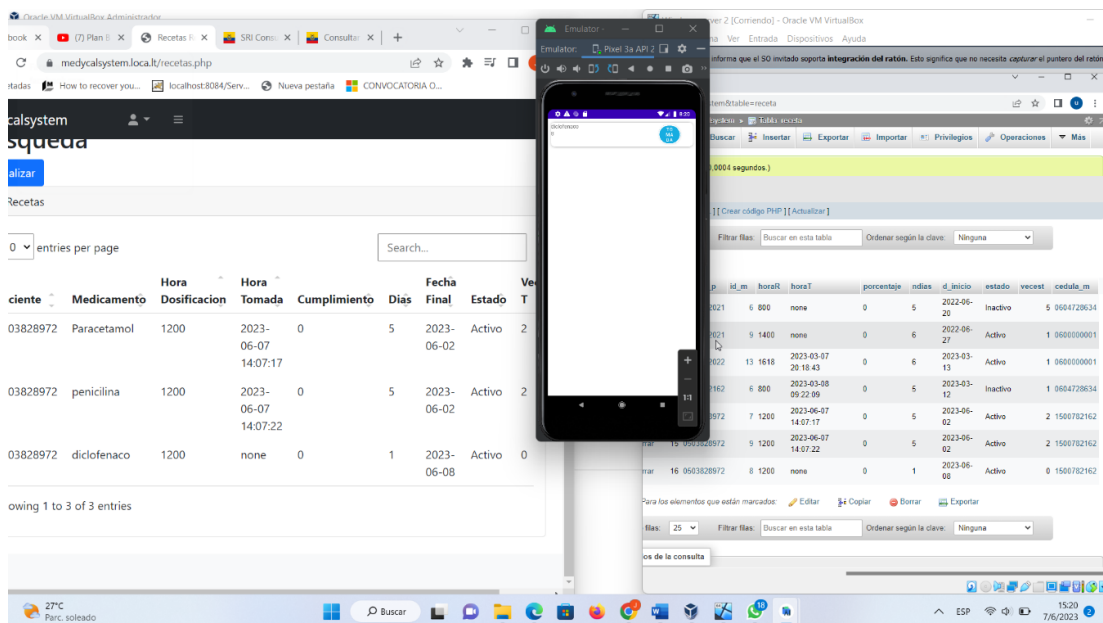


Figura 38: Ejecución del sistema

En la **figura 38** se puede notar que todo el sistema esta interconectado mediante el fog computing en donde el aplicativo móvil es el que emite la información y la que recepta es el aplicativo web.

4.1.3 Detección de errores

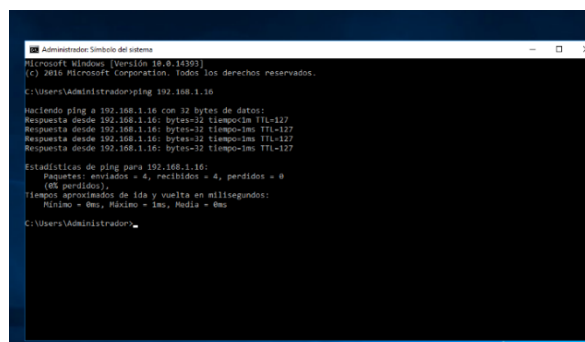
Se puede evidenciar que el sistema es completamente funcional por ende no existen error alguno además que cumple con todos los requerimientos.

4.1.4 Rendimiento

Las pruebas de rendimiento evalúan el desempeño relativo de todos los recursos que se usaron en determinadas condiciones como, por ejemplo:

4.1.4.1 Comportamiento temporal

En la **figura 39** se puede evidenciar él envió de un ping entre el servidor local y un host cualquiera donde se puede apreciar que el 100% de los paquetes es recibido sin novedad y un 0% de paquetes no entregados, de esta forma podemos evidenciar que el aplicativo web y móvil tendrán mejores condiciones para su desarrollo.



```
Administrador Símbolo del sistema
Microsoft Windows [Versión 10.0.14393]
(c) 2016 Microsoft Corporation. Todos los derechos reservados.

C:\Users\Administrador>ping 192.168.1.16

Haciendo ping a 192.168.1.16 con 32 bytes de datos:
Respuesta desde 192.168.1.16: bytes=32 tiempo=1ms TTL=127
Respuesta desde 192.168.1.16: bytes=32 tiempo=1ms TTL=127
Respuesta desde 192.168.1.16: bytes=32 tiempo=1ms TTL=127
Respuesta desde 192.168.1.16: bytes=32 tiempo=1ms TTL=127

Estadísticas de ping para 192.168.1.16:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
            (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 0ms, Máximo = 1ms, Medio = 0ms

C:\Users\Administrador>
```

Figura 39: Envió de ping

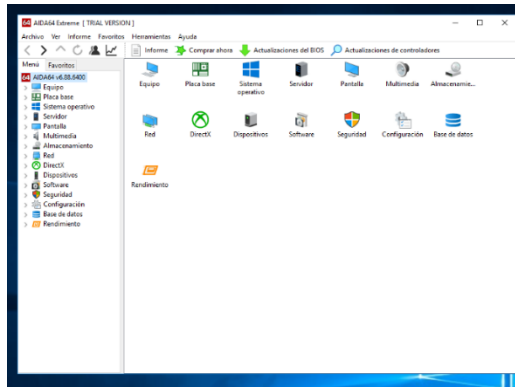


Figura 40: Panel de AIDA64

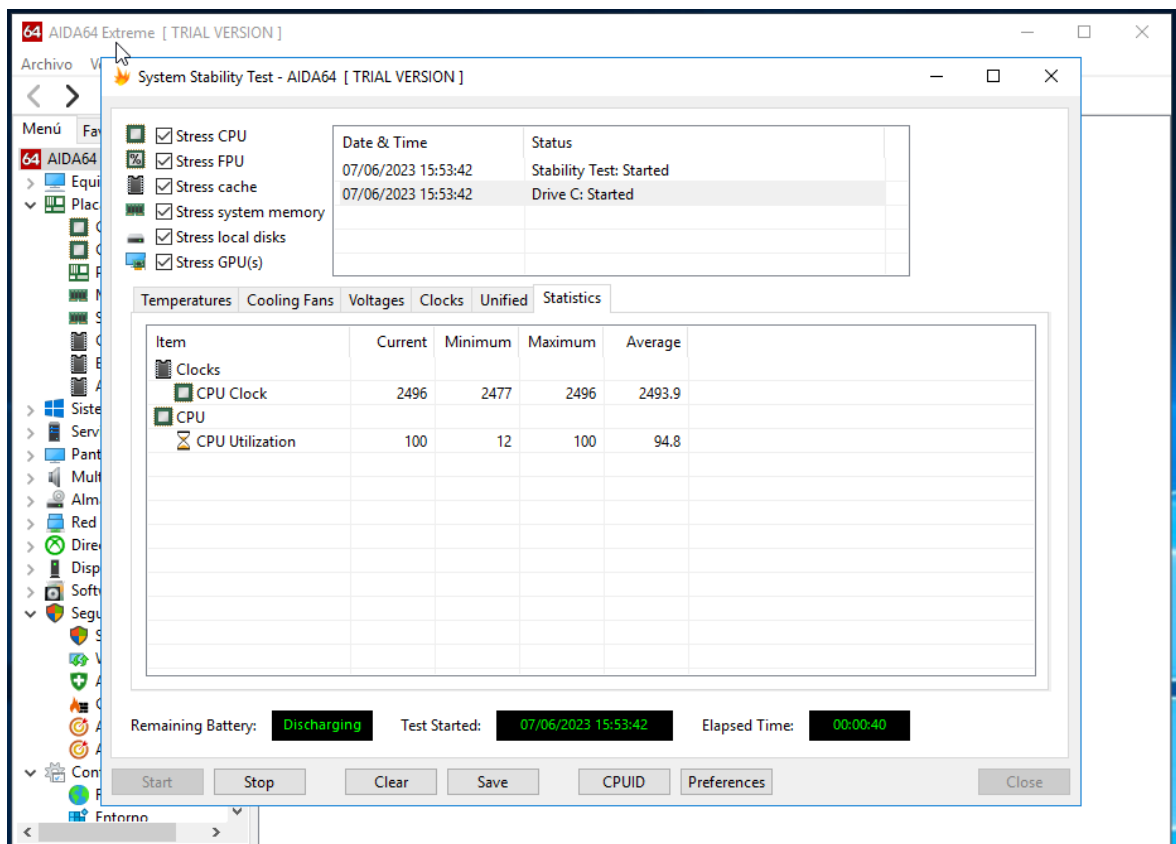


Figura 41: Ejecución de panel de AIDA64

Utilización de recursos: Se estreso el servidor por un tiempo de 15 minutos en el cual se logró determinar temperaturas entre 35 a 45 y sin problemas en el procesamiento de datos puesto que los resultados iban a la mano con la frecuencia del reloj en el procesador y el GPU tiene una frecuencia de 2.5 GHz en el cual se utilizó el 100% del CPU, cabe recalcar que el servidor se encuentra alojado en una máquina virtual con una limitación con la maquina hospedadora, sin embargo los resultados fueron favorables con la utilización de un espacio virtual. Véase en la figura 40 y 41.

4.1.4.2 Capacidad

Tras un estudio pormenorizado utilizando la metodología ágil de programación extrema, se puede concluir que el sistema cumple con todas las especificaciones solicitadas por el cliente y que es 100% vial y fiable para operaciones futuras.

4.1.5 Evaluación del sistema usando jmeter

El primer ensayo tuvo lugar el 10/05/23 a las 16:07 hs. Se utilizaron 100 hilos (threads) con una frecuencia de cada 1 segundo. Los resultados totales obtenidos por la componente "GRAFICA DE RESULTADOS" se presentan en la **tabla 11**.

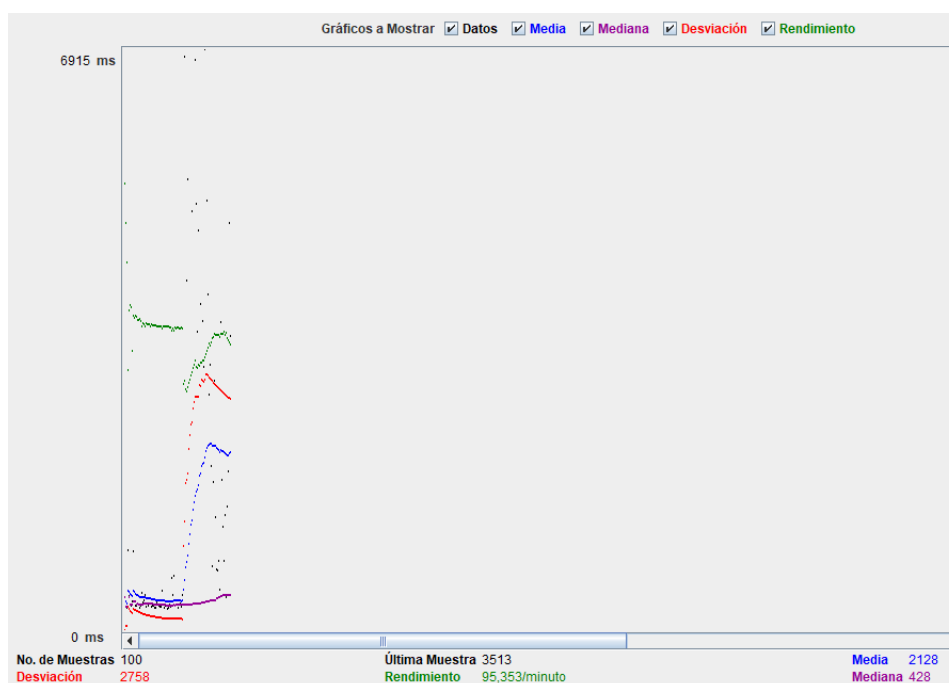


Figura 42: Prueba de Jmeter con 100 clientes

El segundo ensayo tuvo lugar el 11/05/23 a las 15:00 hs. Se utilizaron 250 hilos (threads) con una frecuencia de cada 1 segundo. Los resultados totales obtenidos por la componente "GRAFICA DE RESULTADOS" se presentan en la **tabla 12**.

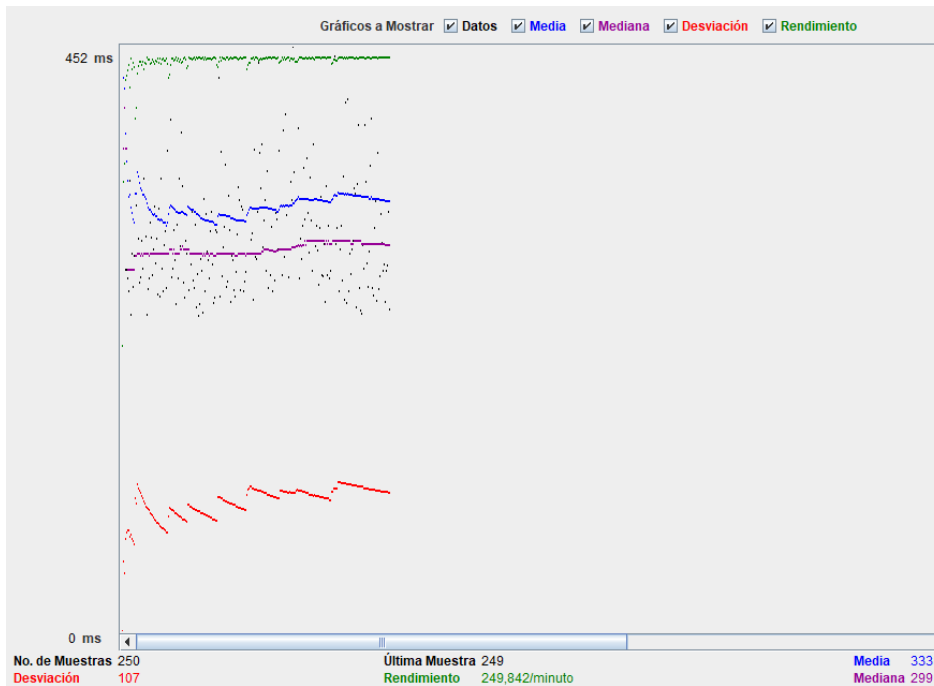


Figura 43: Prueba de Jmeter con 250 clientes

El tercer ensayo tuvo lugar el 12/05/23 a las 12:00 hs. Se utilizaron 500 hilos (threads) con una frecuencia de cada 1 segundo. Los resultados totales obtenidos por la componente "GRAFICA DE RESULTADOS" se presentan en la **tabla 13**.

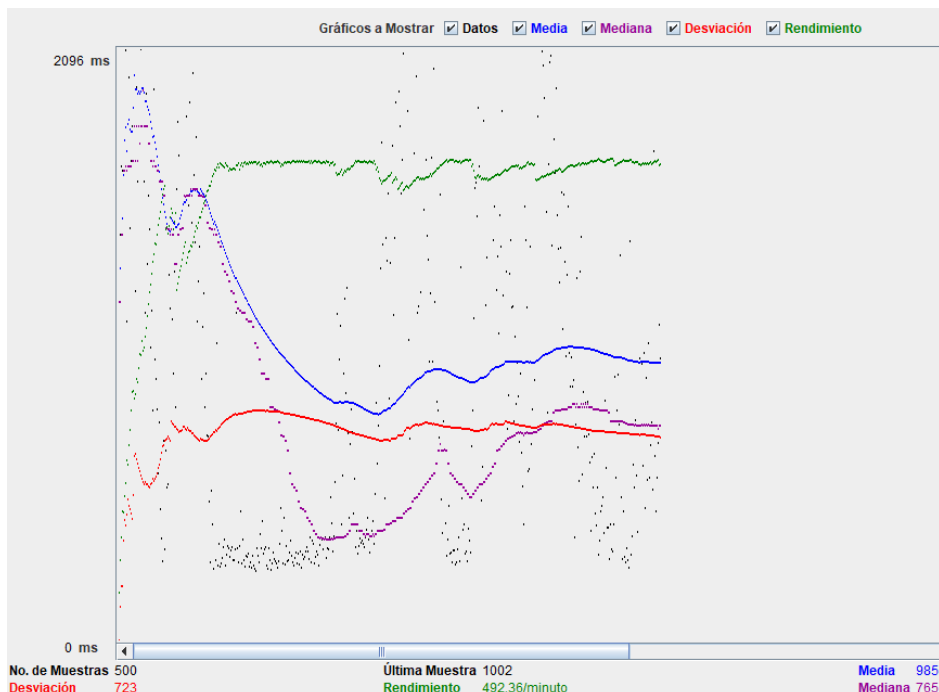


Figura 44: Prueba de Jmeter con 500 clientes

Para poder evaluar el rendimiento del sistema se utilizó el aplicativo Jmeter en el cual se obtuvo las **gráficas 42,43 y 44**, donde los resultados arrojados fueron desde 100 ms hasta 2096 en el máximo de 500 clientes, además podemos apreciar claramente que la pérdida de datos no existe en el último de los casos con 500 clientes el sistema soporta cabe recalcar que en el sistema local mucho depende la capacidad de las características del dispositivo en este caso la máquina virtual esta alojada en una hp core i5 10350h con 16gb de ram y 1 tb de almacenamiento con disco duro y 256 gb de almacenamiento disco solido tipo m.2 que en este caso estamos usando.

Label	# Muestras	Media	Mín	Máx	Std. Dev.	% Error	Rendimiento	Kb/sec	Avg. Bytes
Petición HTTP	100	2128	257	9889	2758,71	0,00%	1,6/sec	5,21	3356,0
TOTAL	100	2128	257	9889	2758,71	0,00%	1,6/sec	5,21	3356,0

Tabla 11: Tiempos de respuesta tabulado en 100 clientes.

Como puede verse en la tabla anterior, el tiempo promedio para acceder a una página es 2,128 segundos, realizándose un total de 100 requerimientos al servidor.

El tiempo total utilizado para los 100 threads es de 212800 milisegundos y el tiempo promedio total requerido por cada thread es de 0.03 en minutos.

Tiempo Total = #Muestras * Media = 100* 2128= 212800 milisegundos.

El tiempo promedio total requerido por cada thread, se puede calcular de la siguiente manera:

$((\text{Tiempo Total} / 1000) / 60) / \text{cantidad de Thread} = ((212800 / 1000) / 60) / 100 = 0,03546$ minutos.

Análisis realizado

Según los resultados arrojados en la **tabla 10**. Se puede esperar que el tiempo de respuesta promedio esté entre 257 ms y 9889 ms segundos para una cantidad de 100 usuarios simultáneos, además se puede evidenciar una tolerancia de 3356, todo esto da a denotar que el rendimiento es mucho mejor utilizando la infraestructura fog computing con recursos de alta gama.

Label	# Muestras	Media	Mín	Máx	Std. Dev.	% Error	Rendimiento	Kb/sec	Avg. Bytes
Petición HTTP	250	333	244	885	107,71	0,00%	4,2/sec	13,65	3356,0
TOTAL	250	333	244	885	107,71	0,00%	4,2/sec	13,65	3356,0

Tabla 12: Tiempos de respuesta tabulado en 250 clientes.

Como puede verse en la tabla anterior, el tiempo promedio para acceder a una página es 2,128 segundos, realizándose un total de 250 requerimientos al servidor.

El tiempo total utilizado para los 250 threads es de 83250 milisegundos y el tiempo promedio total requerido por cada thread es de 0.03 en minutos.

$$\text{Tiempo Total} = \# \text{Muestras} * \text{Media} = 250 * 333 = 83250 \text{ milisegundos.}$$

El tiempo promedio total requerido por cada thread, se puede calcular de la siguiente manera:

$$((\text{Tiempo Total} / 1000) / 60) / \text{cantidad de Thread} = ((83250 / 1000) / 60) / 250 = 0,0003546 \text{ minutos.}$$

Análisis realizado

Según los resultados arrojados en la **tabla 11**. Se puede esperar que el tiempo de respuesta promedio esté entre 244 ms y 885 ms segundos para una cantidad de 250 usuarios simultáneos, además se puede evidenciar una tolerancia de 3356, todo esto da a denotar que el rendimiento es mucho mejor utilizando la infraestructura fog computing con recursos de alta gama.

Label	# Muestras	Media	Min	Máx	Std. Dev.	% Error	Rendimiento	Kb/sec	Avg. Bytes
Petición HTTP	500	2791	238	25526	4048,24	0,00%	6,1/sec	19,99	3356,0
TOTAL	500	2791	238	25526	4048,24	0,00%	6,1/sec	19,99	3356,0

Tabla 13: Tiempos de respuesta tabulado en 500 clientes

Como puede verse, el tiempo promedio para acceder a una página es 2,791 segundos, realizándose un total de 500 requerimientos al servidor.

El tiempo total utilizado para los 500 threads se puede calcular con la siguiente fórmula:
 $\text{Tiempo Total} = \# \text{Muestras} * \text{Media} = 500 * 2791 = 1395500 \text{ milisegundos.}$

El tiempo promedio total requerido por cada thread, se puede calcular de la siguiente manera:

$$((\text{Tiempo Total} / 1000) / 60) / \text{cantidad de Thread} = ((1395500 / 1000) / 60) / 500 = 0,046516 \text{ minutos.}$$

Análisis realizado

Según los resultados arrojados en la **tabla 12**. Se puede esperar que el tiempo de respuesta promedio esté entre 238 ms y 25526 ms segundos para una cantidad de 500 usuarios simultáneos, además se puede evidenciar una

tolerancia de 3356, todo esto da a denotar que el rendimiento es mucho mejor utilizando la infraestructura fog computing con recursos de alta gama.

Tiempos de respuesta				
Clientes	Promedio	Máximo	Mínimo	Error
100	2128	9889	257	0
250	333	885	244	0
500	2791	25526	238	0

Tabla 14: Tiempos de respuesta tabulado

A partir de la información obtenida estresando el servidor por un periodo de 1 minuto, se obtiene información en la cual el mejor promedio o media se da en 250 clientes, esto da a conocer que el 100% de clientes ingreso correctamente, además existen un 0% de error en perdida de datos.

Todo lo antes mencionado aporta significativamente a la tecnología actual y da realce al fog computing para evitar tiempos de latencia, sin embargo, en benchmarks se fuerza aún más al servidor para obtener datos más concisos en futuras pruebas de rendimiento.

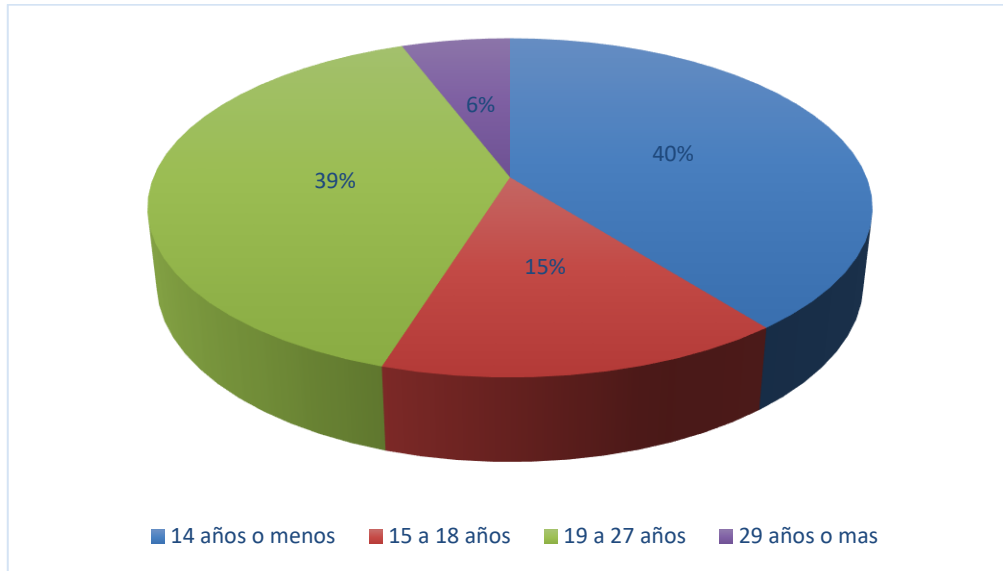
Todas estas pruebas se la hicieron con el fin de dar a conocer que el sistema soporta grandes cantidades de ingreso de clientes y que puede soportar mucho más armando una arquitectura conjunta con otros servidores locales.

4.2 Resultados de encuestas

Resultado de encuesta - Pregunta 1

Seleccione el rango de edad a la que pertenece

35 respuestas



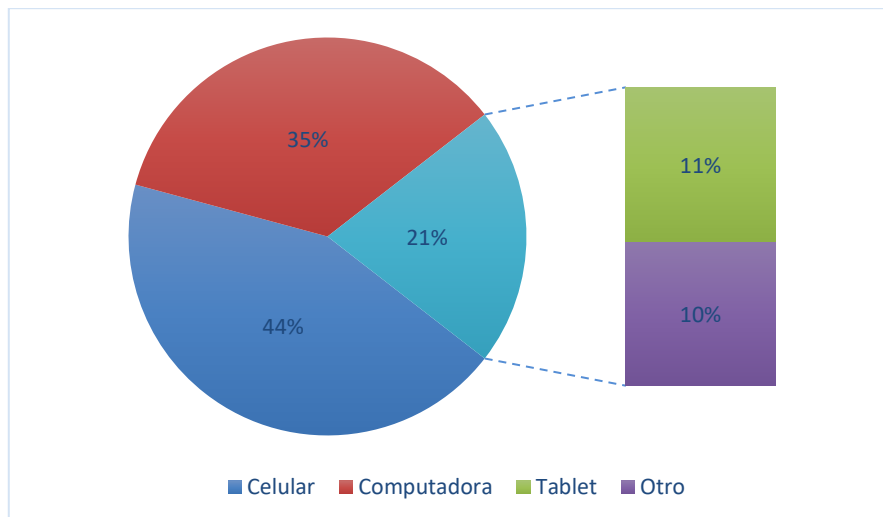
Nota. El gráfico muestra los resultados de la pregunta 1

Fuente: Shiguango- Muñoz, 2023

Resultado de encuesta - Pregunta 2

¿Cuenta con alguno de estos dispositivos inteligentes?

35 respuestas



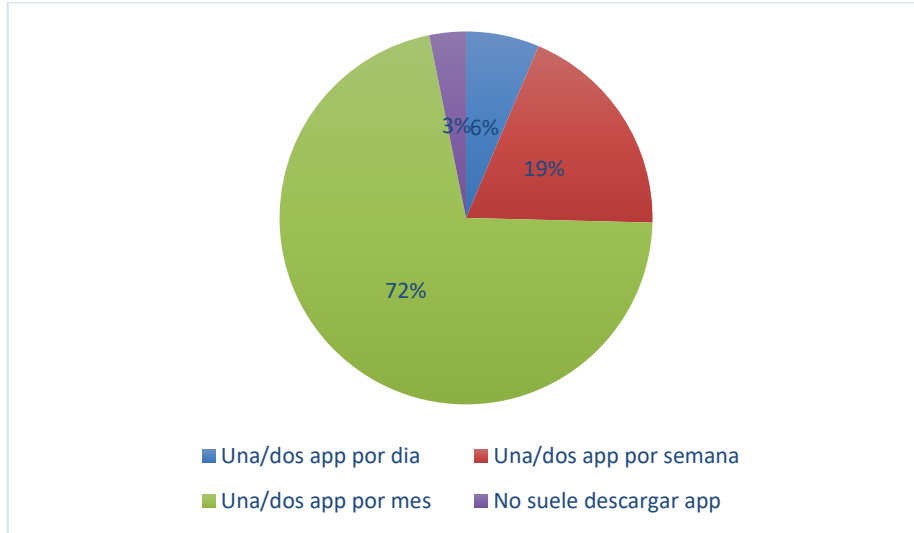
Nota. El gráfico muestra los resultados de la pregunta 2

Fuente: Shiguango- Muñoz, 2023

Resultado de encuesta - Pregunta 3

¿Con que frecuencia descarga aplicaciones(app) en su móvil?

35 respuestas



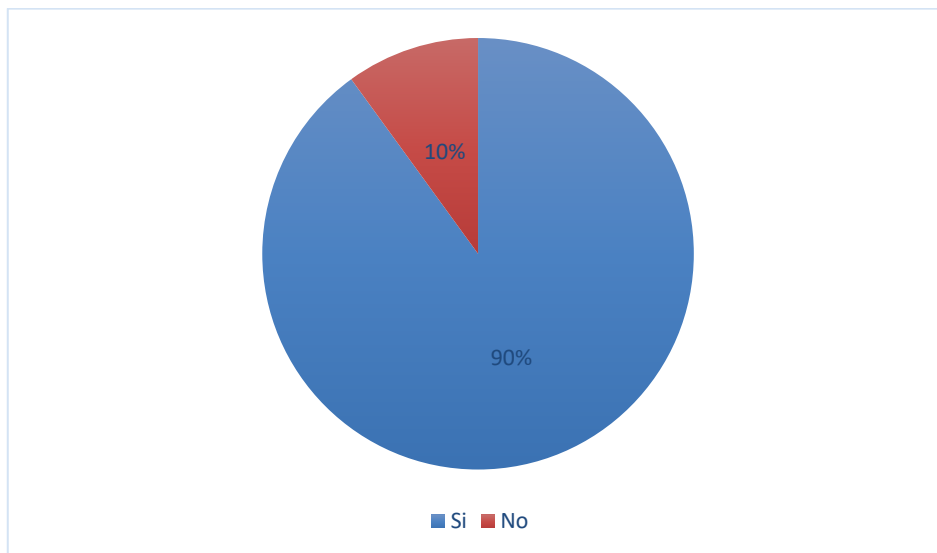
Nota. El grafico muestra los resultados de la pregunta 3

Fuente: Shiguango- Muñoz, 2023

Resultado de encuesta - Pregunta 4

¿Cuenta con servicio de internet o datos móviles?

35 respuestas

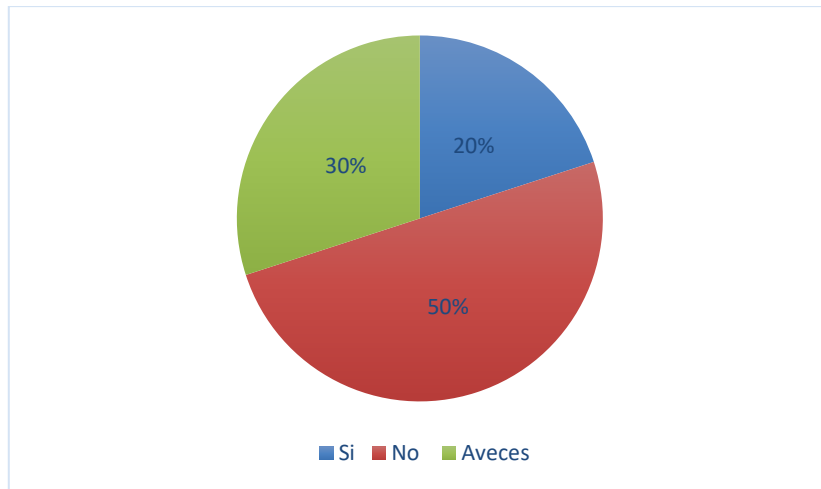


Nota. El grafico muestra los resultados de la pregunta 4

Fuente: Shiguango- Muñoz, 2023

Resultado de encuesta - Pregunta 5

¿Se considera una persona que toma sus medicamentos en las horas establecidas por el doctor?
35 respuestas

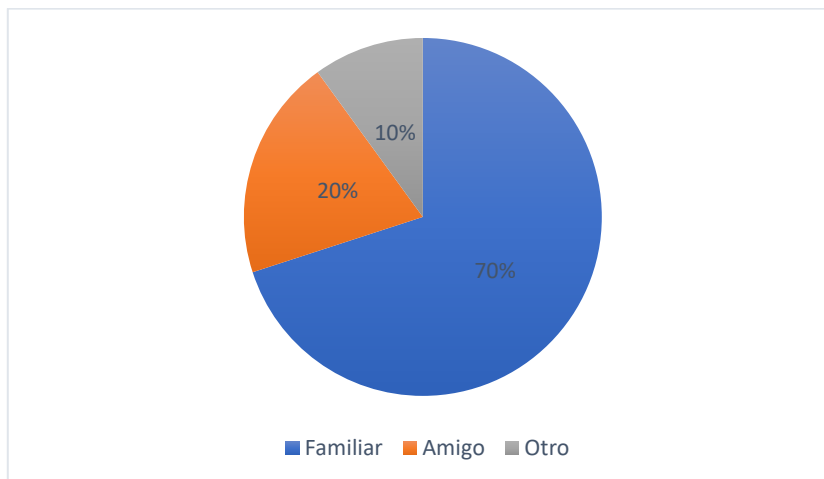


Nota. El grafico muestra los resultados de la pregunta 5

Fuente: Shiguango- Muñoz, 2023

Resultado de encuesta - Pregunta 6

¿Conoce a alguien de su círculo social que se le olvide tomar sus medicamentos enúncielo?
35 respuestas



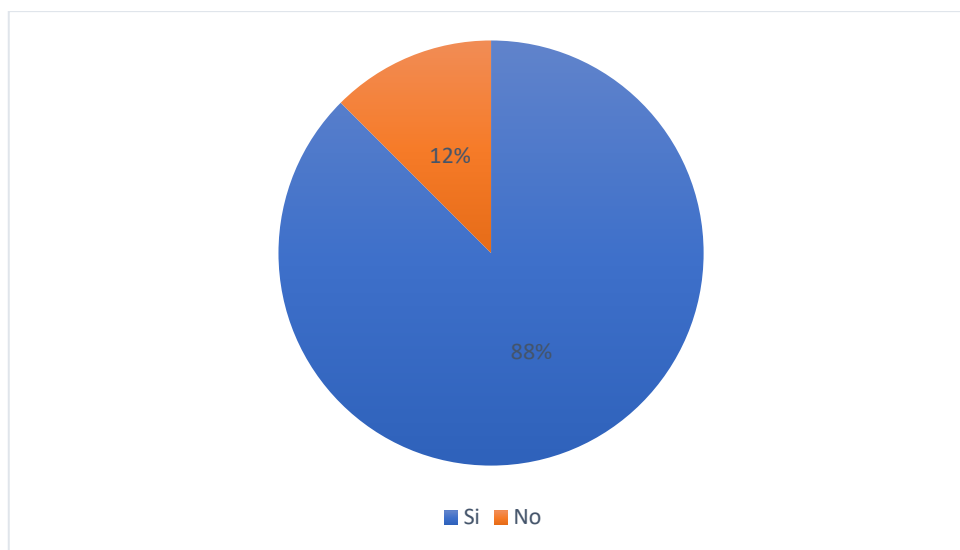
Nota. El grafico muestra los resultados de la pregunta 6

Fuente: Shiguango- Muñoz, 2023

Resultado de encuesta - Pregunta 7

¿Consideraría importante una aplicación móvil para hacer un recordatorio en la toma de sus medicamentos?

35 respuestas



Nota. El gráfico muestra los resultados de la pregunta 7

Fuente: Shiguango- Muñoz, 2023

5. CAPÍTULO V. CONCLUSIONES Y RECOMENDACIONES

5.1 Conclusiones

Fog computing se muestra como una propuesta futurista muy interesante, aunque todavía puede presentar desafíos para los desarrolladores en el Back-End. Estos deben tomar decisiones sobre qué tipo de hosting es más adecuado (VPS, compartido o servidor dedicado), qué bases de datos utilizar (SQL y NoSQL) y determinar cuándo es necesario mantener la integridad de los datos o recopilar datos masivamente. Todos estos planteamientos surgen en sistemas basados en la nube.

En relación con la evaluación del rendimiento, se llevaron a cabo pruebas simuladas en el servidor, y los resultados mostraron un promedio menor en las pruebas con fog en comparación con el enfoque tradicional de Cloud. Por lo tanto, fog se presenta como la opción más adecuada en este contexto.

Aunque los resultados muestran una clara diferencia en los tiempos de respuesta, adoptar una arquitectura completa basada en fog computing puede volverse más complicado debido a los altos costos involucrados en equipos físicos, equipos de desarrollo y seguridad. Además, fog computing no ha sido desarrollado tan a fondo como el modelo de Cloud-Computing, lo que puede llevar a problemas de seguridad significativos. Esta tecnología evita algunas capas de autenticación presentes en Cloud, lo que puede aumentar la vulnerabilidad del sistema. A menos que se desarrollen nuevas herramientas o métodos para garantizar la integridad de los datos y evitar ataques como ataques DDos o SQL Injection que puedan afectar a los equipos físicos, no se recomienda utilizar fog como la opción final para un proyecto de gran envergadura, ya que existe la posibilidad de sufrir los ataques mencionados.

5.2 Recomendaciones

Se aconseja considerar el uso de fog en situaciones en las que el equipo de desarrollo o la empresa puedan afrontar los gastos asociados y dispongan de profesionales especializados en seguridad y encriptación de datos.

Es importante tener en cuenta que fog presenta problemas significativos en términos de seguridad debido a su arquitectura descentralizada con menos capas de protección en comparación con Cloud.

Dado que Cloud es ampliamente preferido para el despliegue de aplicaciones web y móviles, ya que cuenta con diferentes protocolos de encriptación de información, es fundamental tener en cuenta el presupuesto al considerar el uso de fog.

Para tener un control total de sistemas basados en fog, se recomienda utilizar una VPS y basarla en contenedores con Docker, lo que permite optimizar el uso de recursos. Esto incluye asignar adecuadamente el espacio en memoria, la base de datos y el ancho de banda, ya que esto puede afectar significativamente el consumo total de recursos y el presupuesto del proyecto.

Es importante realizar una investigación más exhaustiva sobre fog computing, especialmente considerando todos los equipos físicos necesarios, como routers, clusters, data centers, switch, nodos de fog, entre otros. Estos elementos son fundamentales para cumplir con los criterios establecidos por fog y garantizar el éxito del sistema en su conjunto.

BIBLIOGRAFÍA

- android studio*. (2022). Obtenido de <https://developer.android.com/studio/intro?hl=es-419>
- Anincubator. (2021). *An Incubator*. Obtenido de <https://anincubator.com/que-es-una-aplicacion-movil/>
- anonimo. (s.f.). *Developer android studio*. Obtenido de <https://developer.android.com/studio/intro?hl=es-419>
- Aprendiendo Arduino. (2018). *Arquitectura IoT*. Obtenido de <https://aprendiendoarduino.wordpress.com/2018/11/11/arquitecturas-iot/>
- Bravo, A. R. (2019). *10 códigos HTML básicos para páginas web (con ejemplos)*. Obtenido de <https://mx.godaddy.com/blog/10-codigos-html-para-paginas-web-y-para-que-sirven/>
- Content, R. (2019). *Que es Java*. Obtenido de <https://rockcontent.com/es/blog/que-es-java/>
- Corrales. (2018). *Serie Científica de la Universidad de las Ciencias Informáticas*, 13-23.
- Depor. (2021). Obtenido de <https://depor.com/depor-play/tecnologia/gmail-como-saber-si-leyeron-tu-mail-enviado-aplicaciones-apps-smartphone-celulares-viral-truco-tutorial-estados-unidos-espana-mexico-nnda-nnni-noticia/>
- Desarrollo Web. (2019). Obtenido de <https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/que-es-una-web-app-y-que-clases-hay/>
- developers*. (2018). Obtenido de <https://developer.android.com/studio/intro?hl=es-419>
- Digital Guide Ionos. (2019). *Fog computing*. Obtenido de <https://www.ionos.es/digitalguide/servidores/know-how/fog-computing/>
- Equipo editorial, E. (2021). *Lenguaje de programación*. Obtenido de <https://concepto.de/lenguaje-de-programacion/>. Última edición
- Garcias, F. (2021). *arsys*. Obtenido de <https://www.arsys.es/blog/phpmyadmin>
- hostingred. (2022). *Cloud fundamentos*. Obtenido de <https://www.hostingred.com/cloud/informacion-cloud/>
- Ingertec. (2020). *Ingertec*. Obtenido de <https://ingertec.com/iso-25000-adecuacion-funcional-de-productos-de-software/>
- L., S. (2021). *Rock Content - ES*. Obtenido de [Rock Content - ES: https://rockcontent.com/es/blog/que-es-xml/](https://rockcontent.com/es/blog/que-es-xml/)
- María Gracia. (2022). *Deloitte*. Obtenido de <https://www2.deloitte.com/es/es/pages/technology/articles/IoT-internet-of-things.html>

- Muradas, Y. (2019). *Openwebinars*. Obtenido de <https://openwebinars.net/blog/conoce-las-3-metodologias-agiles-mas-usadas/>
- Muy interesante. (2021). Obtenido de <https://www.muyinteresante.es/tecnologia/articulo/esta-es-la-nueva-funcion-de-dropbox-que-te-facilitara-la-vida-981636966820>
- Normas ISO 25000. (2023). Obtenido de <http://www.iso25000.com/index.php/normas-iso-25000/iso-25010>.
- Perez, A. (2021). *Computacion en la niebla*. Obtenido de https://www.researchgate.net/publication/351564357_Procedimiento_para_la_implementation_de_Computacion_en_la_Niebla_en_Ciudades_Inteligentes_Revista_Ingenieria_Electronica_Automatica_y_Comunicaciones
- Pilataxi, J. L. (2019). *Studocu*. Obtenido de <https://www.studocu.com/ec/document/universidad-politecnica-salesiana/calidad-de-software/como-usar-jmeter-para-pruebas-de-rendimiento-y-carga/8401541>
- Rodríguez, A. (2019). *Deberes del personal de salud y las instituciones prestadoras frente a pacientes con enfermedades crónicas, terminales e incurables respecto a la atención en cuidados paliativos*. Obtenido de <https://repository.upb.edu.co/handle/20.500.11912/4554>
- Sotomayor, S. G. (2022). *IEBS*. Obtenido de <https://www.iebschool.com/blog/que-son-metodologias-agiles-agile-scrum/>
- stackscala. (2021). <https://www.stackscala.com/es/blog/modelos-de-servicio-cloud/>. Obtenido de <https://www.stackscala.com/es/blog/modelos-de-servicio-cloud/>
- Sunyaev, A. (2020). *Springer Link*. Obtenido de https://link.springer.com/chapter/10.1007/978-3-030-34957-8_7
- surveymonkey. (2022). Obtenido de <https://es.surveymonkey.com/mp/sample-size-calculator/>
- Team, A. (2021). *ambit*. Obtenido de <https://www.ambit-bst.com/blog/mhealth-todo-lo-que-debes-saber-sobre-la-salud-m%C3%B3vil>
- Torres, P. (2022). *Computación en Nube*. Obtenido de <http://www2.udec.cl/~aldconcha/cyp/arquitectura.html>
- Ucla. (2020). *Sensores, Iot, salud movil*. Obtenido de <https://oarc.ucla.edu/sensors-iot-mhealth>
- Unitag. (2022). Obtenido de <https://www.unitag.io/es/mobile-websites/what-is-the-difference-between-a-mobile-application-and-a-mobile->



**ACTA DE APROBACIÓN
PERFIL DEL PROYECTO DE INVESTIGACIÓN**

En la ciudad de Riobamba, a los veintidós días del mes de julio de 2022, se reúnen los miembros de la Comisión de Carrera, quienes luego de haber revisado y analizado la petición presentada por los estudiantes **MUÑOZ SARMIENTO ANDERSON JOEL** con CC: 1726009101 y **SHIGUANGO GARCÍA JONATHAN CRISTOFFER** con CC: 1500782162, de la carrera de Ingeniería en Tecnologías de la Información y dando cumplimiento a los criterios metodológicos exigidos, emiten el **ACTA DE APROBACIÓN** del PERFIL DEL PROYECTO DE INVESTIGACIÓN titulado "APLICACIÓN WEB Y MÓVIL DE VIGILANCIA PARA EL CONSUMO DE MEDICAMENTOS EN PACIENTES AMBULATORIOS, UTILIZANDO "FOG COMPUTING", que corresponde al dominio científico "DESARROLLO TERRITORIAL - PRODUCTIVO Y HÁBITAT SUSTENTABLE PARA MEJORAR LA CALIDAD DE VIDA" y alineado a la línea de investigación "Ingeniería Informática".

Mgs. Jorge Delgado
DIRECTOR CARRERA



Mgs. Ana Congacha
MIEMBRO COMISION DE CARRERA

Mgs. Lady Espinoza
MIEMBRO COMISION DE CARRERA

Mgs. Gonzalo Allayca
MIEMBRO COMISION DE CARRERA

Anexo 1: Acta de aprobación de perfil de proyecto de investigación



Riobamba, 22 de julio de 2022
Oficio N° 699-CITI-2022

Ingeniero
Jorge Delgado
TUTOR (A) DEL PROYECTO DE INVESTIGACIÓN
Sres. Anderson Muñoz y Jonathan Shiguango
ESTUDIANTE DE SISTEMAS Y COMPUTACIÓN
Presentes

De mi consideración

Me dirijo a ustedes extendiéndoles un fraterno saludo, a la vez me permito hacerles llegar el acta de aprobación del perfil denominado "APLICACIÓN WEB Y MÓVIL DE VIGILANCIA PARA EL CONSUMO DE MEDICAMENTOS EN PACIENTES AMBULATORIOS, UTILIZANDO "FOG COMPUTING", emitida en la reunión de Comisión de Carrera de Ingeniería en Tecnologías de la Información, efectuada el 21 de julio de 2022, así como también comunicarles que cuentan con la autorización, para que continúen con el desarrollo del trabajo de investigación y ratificar como tutor (a) del proyecto al (a) Ing. Jorge Delgado.

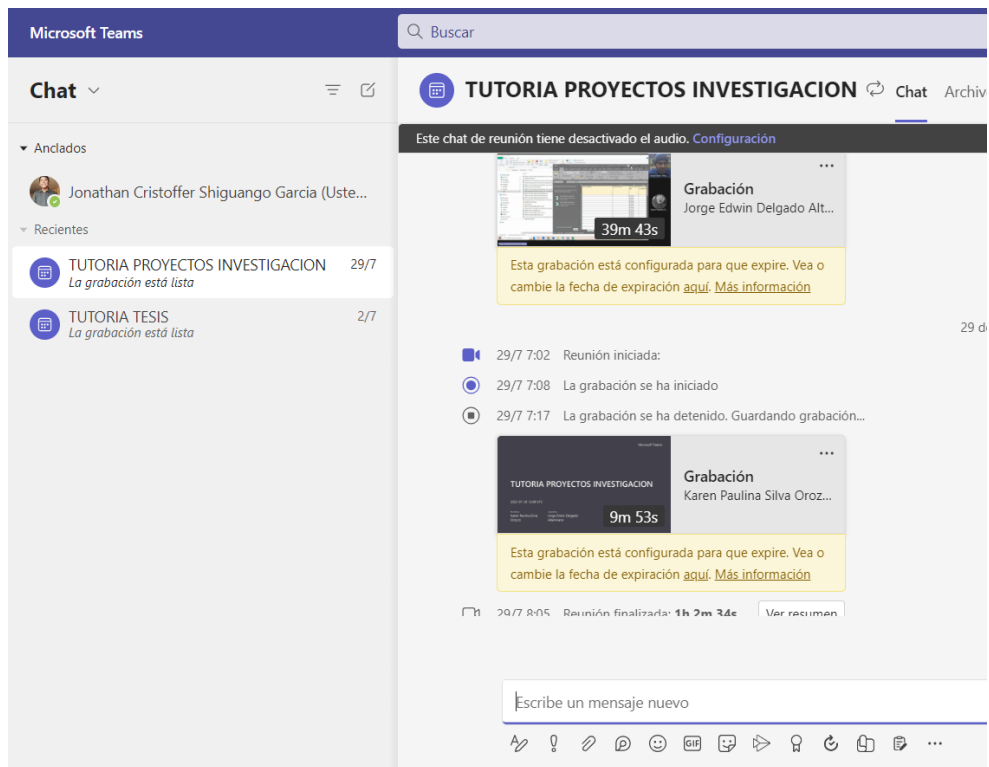
Particular que comunico para el fin legal pertinente.

Cordialmente,



Ing. Jorge Delgado
DIRECTOR DE CARRERA
INGENIERÍA EN TECNOLOGÍAS DE LA INFORMACIÓN
cc. Archivo





Anexo 3: Reuniones virtuales con el tutor de tesis

Entrevista

1. Qué necesidad Ud. prioriza en el sector de la salud?

Para el sector de la salud existen varias deficiencias que deben ser copadas de alguna manera, una de ellas es: como saber tomar los medicamentos, puesto que hay pacientes con graves enfermedades que no saben interpretar las indicaciones, además de eso los horarios muchas de las veces no se la entiende por motivos sea de lectura o escritura, para ello se necesitaría un aplicativo para la toma correcta de medicamentos que envié notificaciones y que pueda monitorear si toma o no en los periodos asignados.

2.Cuál es la utilidad de aplicativos móviles y web en la salud?

La utilidad de estos aplicativos hace que la vida de las personas sea mucho más fácil y eficiente puesto que por medio de estos en los sectores de la salud, educación, industria y mas se ha innovado de manera exponencial, en el sector de la salud ayuda al monitoreo, interpretación y mas de resultados de exámenes y saber signos vitales de manera remota esto aporta a un sondeo en tiempo real de lo que se necesita.

3. Que características tendría el aplicativo que usaría para el sector de la salud?

El aplicativo que necesitaría para el monitoreo tendría una etapa de autenticación además tendría un banner con tres opciones en el cual se puede apreciar el registro y búsqueda de medicamentos, el registro y búsqueda de usuarios y la de receta para poder enviar a los pacientes en diferentes dosis, horarios y asignar a que paciente puede enviar de manera que sepa el tipo de medicamento que le toque y demás características.

4. Ud. considera importante la implementación de un aplicativo de notificaciones?

Considero importante la implementación de un aplicativo de notificaciones por la razón de que hay que estar a la vanguardia de las nuevas tecnologías y usarlas de manera que puedan ser de provecho para la respectiva notificación en tiempo real de la receta que debe tomar y el horario que debe tener.

5. Que impacto tendría un aplicativo para la toma de medicamentos?

Tendría un impacto positivo en la salud de los pacientes puesto que al cumplir con los medicamentos planteados, estos hacen posible que su salud no se deteriore o se controle de manera que será un aporte significativo para el sector de la salud e implementarse en otras áreas o sectores del sector social.

Anexo 4: Resultado de entrevista