

**UNIVERSIDAD NACIONAL JOSÉ MARÍA ARGUEDAS**  
**FACULTAD DE INGENIERÍA**  
**ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS**



Presentado por:

**LENYN ELÍ FLORES BALANDRA**

**MONITOREO Y ALERTA EN PERSONAS CON PROBLEMAS  
CARDIACOS UTILIZANDO WEARABLE EN LA PROVINCIA DE  
ANDAHUAYLAS**

Asesor:

**ING. JUAN JOSÉ ORÉ CERRÓN**

**TESIS PARA OPTAR EL TÍTULO PROFESIONAL DE INGENIERO DE  
SISTEMAS**

**ANDAHUAYLAS - APURIMAC - PERÚ**

**2018**



## APROBACIÓN DEL ASESOR

Quién suscribe:

Ing. Juan José Oré Cerrón por la presente:

### CERTIFICA,

Que, el Bachiller en **Ingeniería de Sistemas**, Lenyn Elí Flores Balandra ha culminado satisfactoriamente el informe final de tesis intitulado: "Monitoreo y alerta en personas con problemas cardiacos utilizando wearable en la provincia de Andahuaylas" para optar el Título Profesional de **Ingeniero de Sistemas**.

Andahuaylas, 24 diciembre del 2018.

  
\_\_\_\_\_  
Ing. Juan José Oré Cerrón  
**Asesor**

  
\_\_\_\_\_  
Bach. Lenyn Elí Flores Balandra  
**Tesista**

# ACTA DE SUSTENTACIÓN DE TESIS



Universidad Nacional José María Arguedas

Identidad y Excelencia para el Trabajo Productivo y el Desarrollo



## FACULTAD DE INGENIERÍA

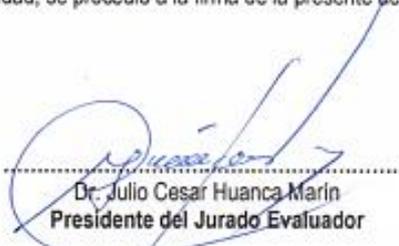
### ACTA DE SUSTENTACIÓN DE TESIS

En la Av. José María Arguedas del Local Académico SL01 (Ccoyahuacho) en el auditorio de la Escuela Profesional de Ingeniería de Sistemas de la Universidad Nacional José María Arguedas ubicado en el distrito de San Jerónimo de la Provincia de Andahuaylas, siendo las 08:00 horas del día 10 de diciembre del año 2018, se reunieron los docentes: Dr. Julio Cesar Huanca Marín, MSc. Iván Soria Solís, MSc. Carlos Yinmel Castro Buleje, en condición de integrantes del Jurado Evaluador del Informe Final de Tesis intitulado: "MONITOREO Y ALERTA EN PERSONAS CON PROBLEMAS CARDIACOS UTILIZANDO WEARABLE EN LA PROVINCIA DE ANDAHUAYLAS", cuyo autor es el Bachiller en Ingeniería de Sistemas **LENYN ELÍ FLORES BALANDRA**, el asesor Ing. Juan José Oré Cerrón, con el propósito de proceder a la sustentación y defensa de dicha tesis.

Luego de la sustentación y defensa de la tesis, el Jurado Evaluador **ACORDÓ:** Aprobar por Unanimidad al Bachiller en Ingeniería de Sistemas **LENYN ELÍ FLORES BALANDRA**, obteniendo la siguiente calificación y mención:

Nota escala vigesimal		Mención
Números	Letras	
17	DIECISIETE	MUY BUENO

En señal de conformidad, se procedió a la firma de la presente acta en 03 ejemplares.

  
.....  
Dr. Julio Cesar Huanca Marín  
Presidente del Jurado Evaluador

  
.....  
MSc. Iván Soria Solís  
Primer Miembro del Jurado Evaluador

  
.....  
MSc. Carlos Yinmel Castro Buleje  
Segundo Miembro del Jurado Evaluador



## APROBACIÓN DEL JURADO DICTAMINADOR

LA TESIS: “MONITOREO Y ALERTA EN PERSONAS CON PROBLEMAS CARDIACOS UTILIZANDO WEARABLE EN LA PROVINCIA DE ANDAHUAYLAS”; para optar el Título Profesional de INGENIERO DE SISTEMAS, ha sido evaluada por el Jurado Dictaminador conformado por:

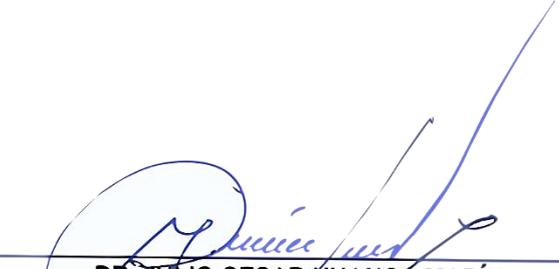
**PRESIDENTE:** DR. JULIO CESAR HUANCA MARÍN

**PRIMER MIEMBRO:** MSC. IVÁN SORIA SOLÍS

**SEGUNDO MIEMBRO:** MSC. CARLOS YINMEL CASTRO BULEJE

Habiendo sido aprobado por UNANIMIDAD, en la ciudad de Andahuaylas el día 10 del mes de diciembre de 2018.

Andahuaylas, 24 de diciembre del 2018.



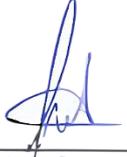
---

**DR. JULIO CESAR HUANCA MARÍN**  
**PRESIDENTE DEL JURADO DICTAMINADOR**



---

**MSC. IVÁN SORIA SOLÍS**  
**PRIMER MIEMBRO DEL JURADO DICTAMINADOR**



---

**MSC. CARLOS YINMEL CASTRO BULEJE**  
**SEGUNDO MIEMBRO DEL JURADO DICTAMINADOR**

## **DEDICATORIA**

El presente trabajo de investigación va dedicado para todos aquellos que desean contribuir al desarrollo cognitivo humano aplicando distintas tecnologías.

Lenyn Elí Flores Balandra

## **AGRADECIMIENTO**

Primeramente, a Dios por ser tan generoso y permitirme elaborar los distintos proyectos que desarrollo en mi vida, a mi papá y mamá que fueron mi más grande soporte y apoyo de vida, mis hermanas que son personas extraordinarias, a mi amigo por su apoyo en todo momento, a mi asesor que fue guía en la presente investigación y a todas aquellas personas que contribuyeron a mi desarrollo personal y profesional.

A mi claustro, por la formación profesional en todas sus dimensiones, a la cual llevare con orgullo en mi pecho GRACIAS QUERIDA UNAJMA.

“emprende con mirada y avanza con visión”

“El desarrollo de una sociedad se basa en la calidad de sus mentes”

Lenyn Elí Flores Balandra

## ÍNDICE

<b>APROBACIÓN DEL ASESOR.....</b>	<b>ii</b>
<b>ACTA DE SUSTENTACIÓN DE TESIS.....</b>	<b>iii</b>
<b>APROBACIÓN DEL JURADO DICTAMINADOR .....</b>	<b>iv</b>
<b>DEDICATORIA .....</b>	<b>v</b>
<b>AGRADECIMIENTO .....</b>	<b>vi</b>
<b>RESUMEN .....</b>	<b>xii</b>
<b>ABSTRACT .....</b>	<b>xiii</b>
<b>CHUMASQA.....</b>	<b>xiv</b>
<b>CAPÍTULO 1: INTRODUCCIÓN .....</b>	<b>1</b>
1.1. Planteamiento del Problema.....	2
1.1.1. Realidad problemática.....	2
1.2. Formulación del problema .....	4
1.3. Objetivos.....	4
1.3.1. Objetivo General.....	4
1.3.2. Objetivos Específicos.....	4
1.4. Justificación.....	4
1.5. Viabilidad de la investigación.....	5
1.5.1. Viabilidad Técnica.....	5
1.5.2. Viabilidad Económica.....	5
1.5.3. Viabilidad Operativa.....	5
1.6. Limitación del estudio.....	5
<b>CAPÍTULO 2: MARCO TEÓRICO .....</b>	<b>6</b>
2.1. Estado del arte .....	6
2.2. Marco conceptual.....	7
2.2.1. Ritmo cardiaco.....	7
2.2.1.1. Componentes del ritmo cardiaco.....	8
2.2.1.1.1. Complejo QRS.....	8
2.2.1.1.2. Onda P.....	8
2.2.1.1.3. Intervalo RR.....	8
2.2.1.1.4. Intervalo PR.....	9
2.2.2. Internet de las cosas(IOT).....	9
2.2.3. Gsm.....	10
2.2.4. Sensor.....	10
2.2.4.1. Sensor Cardiaco.....	11
2.2.4.2. Sensor de pulso cardiaco(SEN-11574).....	11
2.2.5. Arduino.....	12
2.2.5.1. Arduino nano.....	12
2.2.6. Servidor web.....	13
2.2.7. Bootstrap.....	14
2.2.8. ThingSpeak.....	15
2.2.9. Tecnología usable(wearables).....	15
2.2.10. Metodologías ágiles.....	15
2.2.10.1. El Manifiesto Ágil.....	15

2.2.10.2. Programación Extrema.....	17
2.2.10.2.1. Planificación: .....	19
2.2.10.2.2. Diseño: .....	20
2.2.10.2.3. Desarrollo: .....	20
2.2.10.2.4. Pruebas: .....	20
<b>CAPÍTULO 3: MATERIALES Y MÉTODOS.....</b>	<b>22</b>
3.1. Metodología para la implementación del proyecto (Programación Extrema).....	22
3.2. Herramientas Empleadas .....	22
3.2.1. Software.....	22
3.2.1.1. Visual Studio Code.....	22
3.2.1.2. Xampp.....	22
3.2.1.3. Php.....	22
3.2.1.4. Arduino Ide. ....	22
3.2.2. Hardware.....	23
<b>CAPÍTULO 4: RESULTADOS Y DISCUSIÓN .....</b>	<b>24</b>
4.1. Implementación de la metodología ágil Programación Extrema(XP) .....	24
4.1.1. Planificación.....	24
4.1.1.1. Historia de Usuario.....	24
4.1.1.2. Plan de publicaciones (Relese Planning). ....	25
4.1.1.2.1. Primera Iteración.....	26
4.1.1.2.2. Segunda Iteración.....	26
4.1.1.2.3. Tercera Iteración.....	27
4.1.1.3. Velocidad del proyecto.....	28
4.1.1.4. Entregas pequeñas.....	29
4.1.1.5. Programación en parejas.....	29
4.1.1.6. Reuniones diarias.....	29
4.1.2. Diseño.....	30
4.1.2.1. Diseños simples.....	30
4.1.2.2. Glosario de términos.....	40
4.1.2.3. Riesgos.....	40
4.1.2.4. Funcionalidad extra.....	40
4.1.2.5. Refactorizar.....	41
4.1.2.6. Tarjetas CRC.....	41
4.1.3. Desarrollo.....	45
4.1.4. Pruebas.....	57
<b>CONCLUSIONES Y RECOMENDACIONES .....</b>	<b>64</b>
5.1. Conclusiones.....	64
5.2. Recomendaciones .....	65
<b>REFERENCIAS BIBLIOGRÁFICAS.....</b>	<b>66</b>
<b>ANEXOS.....</b>	<b>68</b>
7.1. Historias de usuario .....	68
7.2. Código fuente arduino.....	72
7.3. Funcionamiento INTCAP .....	74

## RESUMEN FIGURAS

FIGURA 1. RITMO CARDIACO NORMAL	7
FIGURA 2. INTERVALO QRS.	8
FIGURA 3. INTERVALO RR.	8
FIGURA 4. INTERVALO PR.	9
FIGURA 5. ELECTROCARDIOGRAMA.	9
FIGURA 6. ARDUINO GSM SHIELD.	10
FIGURA 7. VARIEDAD DE SENSORES CARDIACOS.	11
FIGURA 8. SENSOR DE PULSO CARDIACO.	11
FIGURA 9. LAS PRÁCTICAS SE REFUERZAN ENTRE SÍ.	19
FIGURA 10. FASES DE LA PROGRAMACIÓN EXTREMA.	21
FIGURA 11. HISTORIA DE USUARIO.	25
FIGURA 12. DIAGRAMA DE FLUJO DE DATOS PRINCIPAL.	31
FIGURA 13. DIAGRAMA DE FLUJO DE DATOS: ESTABLECIMIENTO DE LA COMUNICACIÓN.	32
FIGURA 14. DIAGRAMA DE FLUJO DE DATOS: ADQUISICIÓN DEL RITMO CARDIACO.	33
FIGURA 15. DIAGRAMA DE FLUJO DE DATOS: PROCESAMIENTO Y ENVÍO DEL RITMO CARDIACO.	34
FIGURA 16. DIAGRAMA DE FLUJO DE DATOS: MONITOREO.	35
FIGURA 17. DIAGRAMA DE FLUJO DE DATOS: ALERTA.	36
FIGURA 18. MAQUETACIÓN INTERFAZ DE CONTROL DE INGRESO AL SISTEMA.	37
FIGURA 19. MAQUETACIÓN INTERFAZ DE VENTANA PRINCIPAL.	38
FIGURA 20. MAQUETACIÓN MENÚ MONITOREO.	38
FIGURA 21. MAQUETACIÓN MENÚ REPORTE.	39
FIGURA 22. MAQUETACION MENÚ ACTUALIZACIÓN.	39
FIGURA 23. MAQUETACIÓN MENÚ REPORTE.	41
FIGURA 24. PATRÓN MODELO VISTA CONTROLADOR.	45
FIGURA 25. CONTROL DE INGRESO AL SISTEMA.	46
FIGURA 26. INTERFAZ PRINCIPAL SISTEMA CARDIACO.	46
FIGURA 27. ENVÍO DE RITMO CARDIACO AL SERVIDOR THINGSPEAK.	48
FIGURA 28. MENÚS DEL SISTEMA CARDIACO.	49
FIGURA 29. CONEXIÓN FÍSICA DE LOS COMPONENTES DEL WEARABLE ELABORADO EN FRITZING.	50
FIGURA 30. CONEXIÓN Y ENVÍO DE BPM DEL GSM CON THINGSPEAK.	51
FIGURA 31. VISUALIZACIÓN DEL RITMO CARDIACO EN THINGSPEAK.	52
FIGURA 32. VISUALIZACIÓN DEL RITMO CARDIACO EN EL SISTEMA WEB DE MONITOREO.	52
FIGURA 33. ACTUALIZACIÓN DE DATOS DEL PACIENTE.	53
FIGURA 34. HISTORIAL DE OBSERVACIONES DEL PACIENTE.	54
FIGURA 35. ADMINISTRADOR DE LA BASE DE DATOS.	54
FIGURA 36. INTEGRACIÓN DE LOS COMPONENTES DEL WEARABLE EN EL GORRO.	55
FIGURA 37. RECUBRIMIENTO INTERNO DE LA GORRA.	56
FIGURA 38. SISTEMA WEB RESPONSIVO.	57
FIGURA 39. CÓDIGO FUENTE DE LA ALERTA.	57

## RESUMEN TABLAS

TABLA 1. ESPECIFICACIONES TÉCNICAS DEL ARDUINO NANO.....	12
TABLA 2. TIPOS DE SERVIDOR. ....	13
TABLA 3. CUADRO DE ITERACIONES.....	28
TABLA 4. CUADRO DE ENTREGAS.....	29
TABLA 5. TARJETA CRC: CLASE USUARIO.....	42
TABLA 6. TARJETA CRC: CLASE PACIENTE. ....	42
TABLA 7. TARJETA CRC: CLASE INGRESO AL SISTEMA.....	42
TABLA 8. TARJETA CRC: CLASE CAPTURA DEL RITMO CARDIACO.....	43
TABLA 9. TARJETA CRC: PROCESAMIENTO DE LOS VALORES. ....	43
TABLA 10. TARJETA CRC: CLASE ENVIÓ DE DATOS AL SERVIDOR. ....	43
TABLA 11. TARJETA CRC: CLASE ALERTA.....	44
TABLA 12. TARJETA CRC: CLASE MONITOREO. ....	44
TABLA 13. PRUEBA DE ACEPTACIÓN: ACCESO AL SISTEMA CON LOS DATOS CORRECTOS. ....	58
TABLA 14. PRUEBA DE ACEPTACIÓN: CORRECTO FUNCIONAMIENTO DEL WEARABLE.....	59
TABLA 15. PRUEBA DE ACEPTACIÓN: CORRECTO ENVÍO DEL RITMO CARDIACO.....	59
TABLA 16. PRUEBA DE ACEPTACIÓN: CORROBORACIÓN EN LA FACILIDAD DE USO. ....	60
TABLA 17. PRUEBA DE ACEPTACIÓN: CORRECTO MONITOREO Y VISUALIZACIÓN DEL RITMO CARDIACO.....	60
TABLA 18. PRUEBA DE ACEPTACIÓN: MANEJO DE LOS DATOS DEL CLIENTE. ....	61
TABLA 19. PRUEBA DE ACEPTACIÓN: ALERTA DE PROBLEMAS CARDIACOS.....	62
TABLA 20. PRUEBA DE ACEPTACIÓN: CORROBORACIÓN DE LA VISUALIZACIÓN EN DISTINTOS DISPOSITIVOS..	62

## RESUMEN GRÁFICOS

GRÁFICO 1. PRIMERA ITERACIÓN.....	26
GRÁFICO 2. SEGUNDA ITERACIÓN.....	27
GRÁFICO 3. TERCERA ITERACIÓN.....	28

## RESUMEN

Los problemas cardiacos cada vez incrementan la tasa de morbilidad y esto se dice a razón de que 17 millones de personas mueren al año a causa de enfermedades referidas al ámbito cardiaco, y es alarmante aún más cuando para los próximos años crecerá la morbilidad.

Monitoreo y alerta en personas con problemas cardiacos utilizando wearable en la provincia de Andahuaylas tiene como objetivo principal monitorear las anomalías cardiacas que pueden surgir repentinamente en personas que tienen antecedentes de este tipo de enfermedades; con la creación de un wearable implementado en un gorro denominado "INTCAP" y su respectivo sistema web de monitoreo, que se podrá visualizar desde cualquier lugar con acceso a internet ya sea desde una computadora, un Smartphone o una Tablet, así también a través de este dispositivo se puede realizar alertas al cardiólogo cuando exista anomalías en el correcto funcionamiento del corazón para su inmediato actuar y toma de decisiones en función a las variaciones del ritmo cardiaco que existiese.

Para el correcto desarrollo de la presente investigación se utilizó la metodología ágil de desarrollo "Programación Extrema(XP)" y el modelo arquitectónico de desarrollo "Modelo Vista Controlador(MVC)".

Se logró la correcta integración del arduino nano con el módulo Shield SIM900, Sensor cardiaco y la plataforma web de monitoreo, que en conjunto logran realizar el monitoreo y la alerta.

Este proyecto no es cerrado ya que se pueden incrementar mayores funcionalidades y módulos que ayuden a precisar con mayor exactitud señales vitales que se consideren primordiales ya que la plataforma hardware que se utilizo es de código abierto al igual que los distintos componentes.

**Palabras claves:** Arduino nano, módulo Shield SIM900, Sensor cardiaco, sistema web, Programación Extrema, Modelo Vista Controlador.

## **ABSTRACT**

The Cardiac problems increasingly increase the rate of morbidity and this is said because 17 million people die each year from diseases related to the cardiac environment, and it is even more alarming when morbidity will grow for the next few years.

Monitoring and alerting in people with heart problems using wearable in the province of Andahuaylas has as main objective to monitor cardiac abnormalities that may arise suddenly in people who have a history of this type of diseases; with the creation of a wearable implemented in a hat called "INTCAP" and its respective web monitoring system, which can be viewed from anywhere with Internet access either from a computer, a Smartphone or a Tablet, as well as through This device can be alerted to the cardiologist when there are abnormalities in the correct functioning of the heart for immediate action and decision making depending on the variations in heart rate that exist.

For the correct development of this research we used the agile development methodology "Extreme Programming (XP)" and the architectural model of development "Model View Controller (MVC)".

The correct integration of the arduino nano was achieved with the Shield SIM900 module, Cardiac Sensor and the monitoring web platform, which collectively manage monitoring and alerting.

This project is not closed because it can increase greater functionalities and modules that help to more accurately specify vital signs that are considered essential since the hardware platform that was used is open source as well as the different components.

Keywords: Arduino nano, Shield module SIM900, Cardiac sensor, web system, Extreme Programming, Model View Controller.

## CHUMASQA

Puywankunapa, sunqukunapa sasachakuyninkunan unquyninkunan sapa punchaw, sapa kuti yapakamuchkan, chaymi wañuyman chayachiwachkanchik, chaysi chiqap kanman chunka qanchisniyuq hunu runakuna wañun huk watallapi kay puywan unquy hapiwasqanchikwan, hinallataqmi niwanchik siqiq watakunapiqa haswantas wiñanman kay puywan unquykunawan wañuykuna.

Antawaylla llaqtanchikpi kay “WEREABLE” nisqan yanapakuyninwan llapa puywan nanay unquyta allinta qatipananchik; “KAY LLANKAYPA ALLIN TARIYNINQA: QATIPAYMI LLAPA RUNAKUNAPA PUYWAN UNQUYKUNA ÑAWPAQ HAPICHIKUQKUNA MANAÑA ASWANTA MIRANANPAQ, WIÑANANPAQ”. Huk chukupi unanchasqa, rurasqa kay “WEREABLE” sutinchasqa “INTCAP”; Chayllamantaq kanqa huk qatipaquiyoq, qawaqniyuq maylaw pachamantapas kay: internet, computadora, Smartphone chaynallataq Tablet nisqanmantapas; Kaykuna willakamunqa, huk yachaqman, puywan unquykunamanta runakuna sasachakuypi tarikuptin chayman hina allin pachallapi asuykuspa qawarinanpaq llamirinanpaq allinchus manachus kachkan, ichaqa pantachkanchus puywannin.

Kay sumaq hatun chuya yuyaymanay rurayninpiqa, allin llankayninpi llusqinanpaqa, rurakurqan allin pispita churapakuywanmi, yuyaymanaywanmi, kay programación extrema (XP) chaynallataq allin qawachikuy arquitectónico ñawpaqman puririy modelo vista controlador (MVC).

Hispirunmi allin away, hapichinakuy, kay llankaywan: ARDUINO NANO nisqanwan, hinallataq kay MODULO SHIELD SIM900 SENSOR CARDIACO qatipaquin WEB MONITOREO, kuskalla llapallanku allinta hapiykanakuspa qatipanqaku hinallataq willakunqaku, puywankunapa unquyninmanta sasachakuyninkunamanta.

Kay llankaykunaqa manan wichqasqachu, aswanqa yapaswanmi ima yuyaymanaynichiktapas, ichaya kanmanpas huk aswan allin, allin qatipaypas yapaykunapas, aswan allin chiqap yachayman, willakuymán, hampiyman chayananchikpaq; Kay away HARDWARE nisqanpiqa rurakurqa kichasqa código nisqanwanmi, hinallataq wakin llankaqmasinkunapas kichasqataqmi kachkan.

Chuya qillqakuna: Arduino nano, módulo Shield SIM900, Sensor cardiaco, sistema web, Programación Extrema, Modelo Vista Controlador.

## CAPÍTULO 1: INTRODUCCIÓN

Al 2018 la salud es un pilar fundamental en la sociedad, y el uso de la tecnología en este sector es cada vez mayor, reemplazando incluso funciones de los distintos profesionales que ejercen su profesión en este campo; las enfermedades cardiacas son unas de las principales causas de mortalidad en el mundo y las personas buscan alternativas y soluciones para poder controlar este tipo de enfermedades, por otro lado los wearables son dispositivos que en esta última década está tomando gran relevancia por su valor de uso y en combinación con el Internet de las cosas son una herramienta potencialmente útil.

A partir de este problema y el impacto de estas herramientas y tecnologías surge la idea de desarrollar un dispositivo (wearable) que sea capaz de monitorear en tiempo real el ritmo cardiaco y a su vez alertar al cardiólogo correspondiente cuando existan anomalías con el correcto funcionamiento del corazón para su inmediata asistencia.

La programación web no resulta difícil ya que existe abundante información y el conocimiento necesario para poder desarrollarlo sin complicación alguna y este es el mismo caso para la programación en arduino que con sus librerías y su constante actualización ayudan cada vez más al fácil desarrollo de artefactos con este dispositivo, pero eso si la dificultad radica en la integración de los distintos módulos y su compatibilidad en el espacio geográfico donde utiliza. La metodología empleada para desarrollar la presente investigación fue la Programación Extrema (XP) que gracias a sus fases permitió el fácil desarrollo del proyecto el cual a través de sus iteraciones nos ayudó a desarrollar los menores errores posibles.

Por lo consiguiente, el presente informe está constituido de la siguiente manera:

Introducción: Aquí se encuentra el planteamiento del problema, la realidad problemática donde se describe los problemas localizados así mismo se encuentran los objetivos, la justificación, la viabilidad y las limitaciones del estudio.

Marco teórico: En aquí encontraremos los antecedentes para la presente investigación así también todos los conceptos necesarios debidamente citados para poder desarrollar con normalidad en todas sus dimensiones la presente investigación.

Materiales y métodos: En este capítulo se detallan los distintos materiales a utilizar y la metodología de desarrollo ágil "Programación Extrema" que trabajará en conjunto con la arquitectura Modelo Vista Controlador(MVC).

Resultados y Discusión: La implantación de la metodología y la arquitectura para desarrollar el sistema web de monitoreo y el wearable.

Y por último encontraremos las conclusiones, recomendaciones y anexos.

## **1.1. Planteamiento del Problema**

### **1.1.1. Realidad problemática.**

En la última década se viene viviendo una preocupación en el ámbito mundial por la tasa de morbilidad a causa de la creciente problemática en el control de personas que tienen problemas cardiacos, y es más preocupante aun cuando el 50% de estas personas que sufren de problemas cardiacos no presentan ningún síntoma previo al ataque que sufren de este.

Cada vez es más latente este problema de carácter mundial y las distintas organizaciones como la Organización Mundial de la Salud (OMS), la WORLD HEART FEDERATION entre otros están en búsqueda de soluciones para poder disminuir la tasa de morbilidad a causa de esta enfermedad no transmisible, y así nos manifiesta. Las enfermedades cardiacas todavía se mantienen como la primera causa de muerte en todo el mundo, muy por encima del cáncer, las enfermedades respiratorias y la diabetes. Alrededor de 17 millones de personas fallecen al año por esas afecciones y se estima que, en el 2023, el número crecerá a más de 23 millones(Infobae,2016).

Esta situación es un gran problema para el sector salud. Por otro lado, en función a esta problemática también se manifiesta la Organización Mundial de la Salud, manifiesta que las Enfermedades Cardiovasculares (ECV) son la principal causa de muerte en todo el mundo. Cada año mueren más personas por ECV que por cualquier otra causa. Se calcula que en 2012 murieron por esta causa 17,5 millones de personas, lo cual representa un 31% de todas las muertes registradas en el mundo. De estas muertes, 7,4 millones se debieron a la cardiopatía coronaria, y 6,7 millones, a los Accidente Cerebrovascular (AVC). Más de tres cuartas partes de las defunciones por ECV se producen en los países de ingresos bajos y medios. De los 16 millones de muertes de personas menores de 70 años atribuibles a enfermedades no transmisibles, un 82% corresponden a los países de ingresos bajos y medios y un 37% se deben a las ECV(OMS,2017).

Los problemas cardiacos en su mayoría afectan a países con recursos económicos bajos y medios por esta razón es que ya es un problema muy latente en la sociedad. Sin embargo, en américa, el problema cardiaco no es ajeno, ya que en esta parte del mundo se da la mayor tasa de morbilidad a causa de problemas cardiacos, la American Heart Association en su resumen de estadísticas del 2017, nos menciona que alrededor de 92.1 millones de personas americanas sufren de enfermedades cardiacas, de los cuales 47,7 % son mujeres y 46% son varones, esta es una tasa realmente alarmante para el bienestar social (Benjamin et al., 2017).

En el caso de Perú según manifiesta en el portal de VITAL.

El 15% de las muertes son por causas de enfermedades cardiovasculares, y esto es más peligroso aún porque se sabe que el 50 % de las mencionadas muertes no presentan síntomas antes del ataque mortal(VITAL,2014), esto se ve muy preocupante para la realidad en la salud peruana, igualmente el Ministerio de salud nos dice. Las enfermedades cardíacas constituyen una de las causas principales de discapacidad y muerte prematura en todo el mundo. En Perú la tasa de mortalidad por enfermedad isquémica de corazón, que es la que provoca una angina de pecho o un infarto, es de 28.77 por ciento, y un dato más resaltante es que 3 personas fallecen cada hora en el Perú por enfermedades cardíacas (MINSA, 2013).

El problema no solo radica en la cantidad personas que sufren de la mencionada enfermedad también el gasto que debe realizar la nación para poder contrarrestar esta mencionada enfermedad y así lo dicen los estudios realizados, según la siguiente nota de prensa: Un reciente estudio de la firma de auditoría internacional Deloitte preparado para América Latina revela que solo en Perú se pierden S/2,900 millones al año por enfermedades cardíacas , tales como la Insuficiencia Cardíaca, el Infarto de Miocardio, la Fibrilación Auricular y la Hipertensión. Esto se debe a los gastos dedicados a la atención de 3,2 millones de personas que padecen estas patologías: la mortalidad prematura, los costos para el cuidador y la pérdida de productividad, ausentismo o menor participación de la fuerza de trabajo y costos de atención informal (Perú21, 2017).

Al 2018 la tecnología juega un papel muy importante en muchos por no decir todos los ámbitos existentes del mundo y la salud es un área donde la tecnología tiene una gran importancia es así que en Europa se reduce enormemente la tasa de morbilidad por causa de enfermedades cardíacos gracias al uso de tecnologías orientadas a este caso, desde las operaciones hasta el control de las distintas personas que tienen problemas cardíacos, y es así que gracias a las distintas tecnologías desarrolladas en la última década la mejora en la cantidad de morbilidad va descendiendo con respecto a las enfermedades cardíacos aunque por malas prácticas en el cuidado de la salud de las personas con relación a la colaboración tecnológica, el crecimiento en esta enfermedad que se estima al 2021 es mayor, lo ideal sería que los índices de morbilidad con relación a los problemas cardíacos decrementsen para las próximas décadas.

Pero viendo la realidad en la cual estamos en el 2018, y en específico en el Perú no se le da la importancia debido a este tema y eso que es la primera causal de muertes a nivel mundial. Se tiene potencial en capacidad humana, recursos y demás elementos que se pueden dar uso para la mejora en aspectos de salud y en específico con enfermedades cardíacos, existen distintas tecnologías que están al alcance de las

personas con intereses en poder solucionar este problema, el hardware libre, software libre, y el conocimiento que se puede encontrar en el internet para poder frenar en porcentajes considerables esta enfermedad.

Otro problema es el desinterés por parte de las personas conocedoras sobre los aspectos de tecnologías y el poco interés sobre valorar las nuevas tecnologías emergentes como son las tecnologías usables también llamadas wearables y aun peor en la provincia de Andahuaylas que son escasa las personas que conocen del tema, y el gran impacto que tendría si se diera el buen uso de este tipo de tecnologías.

## **1.2. Formulación del problema**

¿Cómo monitorear los problemas cardiacos haciendo uso de wearables en la provincia de Andahuaylas?

## **1.3. Objetivos**

### **1.3.1. Objetivo General.**

Desarrollar un wearable para realizar la monitorización y alerta en tiempo real en personas con problemas cardiacos en la provincia de Andahuaylas.

### **1.3.2. Objetivos Específicos.**

- Realizar la planificación del wearable y el sistema web de monitoreo con relación a los problemas cardiacos.
- Diseñar del wearable y del sistema de monitoreo.
- Desarrollar el wearable y el sistema web de monitoreo de problemas cardiacos.
- Realizar las pruebas del sistema web de monitoreo y wearable.

## **1.4. Justificación**

La presente investigación logrará una gran relevancia por el énfasis sobre los problemas cardiacos que es el problema principal de mortandad en el mundo, en la última década los problemas cardiacos vienen creciendo pese a que la tecnología avanza para poder frenar este tipo de problemas, la solución que se presentará contribuirá en ayudar a controlar lo problemas cardiacos en la provincia de Andahuaylas, los problemas de la salud son aspectos muy álgidos y por ende se le debe de dar la relevancia correspondencia.

En el seguro social de salud de la provincia de Andahuaylas solo se cuenta con un cardiólogo que el trabajo que brinda es quincenal de esta forma el control de personas con problemas cardiacos no es la más óptima.

Pese a que en el 2018 ya existen distintos métodos para poder controlar a personas con problemas de salud que puede ser la telemedicina y los wearables que solo controlan de forma local y no existe comunicación con el médico responsable de la persona que la pueda estar usando así también, por desconocimiento y costo elevado no es fácil de implementar en esta parte del país.

## **1.5. Viabilidad de la investigación**

### **1.5.1. Viabilidad Técnica.**

Se cuenta con la capacidad humana, los recursos tecnológicos los cuales son: arduino nano, sensor cardiaco, GSM Shield, también se cuenta con servicios como internet, telefonía; la metodología de Programación Extrema.

Es integrable el arduino nano con el GSM Shield y el sensor en una gorra con su respectiva batería y se procederá al envío de la información a una plataforma web de alerta y monitoreo, con todo ello el presente proyecto es viable técnicamente.

### **1.5.2. Viabilidad Económica.**

Es viablemente económico por que el monto que demandara el presente proyecto es de S/. 2459.38 soles el cual es accesible para el investigador, en el caso que la investigación tenga un gran impacto la patente y comercialización del producto será óptima; el beneficio se calculará durante la ejecución del presente proyecto.

### **1.5.3. Viabilidad Operativa.**

La presente investigación es operativamente viable porque la creación del wearable será de uso para las personas con problemas cardiacas, el uso será muy fácil e intuitivo, porque es prácticamente ponerse una gorra en la cabeza y recargar la batería del wearable cada vez que lo requiera, el uso de la plataforma lo podrán realizar los interesados como son los familiares y el cardiólogo responsable, incluso facilitará al monitoreo de las personas con este tipo de enfermedades, se les explicará y capacitará a los usuarios para el uso de este aparato a crear.

## **1.6. Limitación del estudio**

- Variabilidad en el precio de los aparatos electrónicos.
- Interferencia electromagnética en personas con variabilidad de carga.
- Políticas dentro del establecimiento de salud para el desarrollo de la investigación.
- Interferencia en la señal para la notificación de los respectivos controles.
- El presente proyecto se desarrollará en la provincia de Andahuaylas donde exista cobertura de datos móviles.

## CAPÍTULO 2: MARCO TEÓRICO

### 2.1. Estado del arte

En un contexto mundial al 2018 se vino desarrollando distintas investigaciones para poder realizar el control del ritmo cardiaco con varias formas de realizar la comunicación de los datos obtenidos con los centros de salud, personas interesadas, etc.

Es así que en Catalunya-Barcelona se hizo el “Diseño, desarrollo y test de un prototipo de pulsera para adquirir el electrocardiograma y la onda de pulso” perteneciente a Alberto Barea Cañizares que llega a las siguientes conclusiones después de desarrollar la investigación:

(Barea,2016) La calidad en la lectura de la señal de ECG ha sido mejorada al cambiar el electrodo a la cara interna de la muñeca, evitando de esta manera el contacto con el pelo del brazo. También se ha vencido el inconveniente que ha supuesto la utilización de electrodos secos para adquirir la señal de ECG superando los problemas que han surgido y proponiendo posibles soluciones.

Respecto a la señal de pulso, cabe decir que la lectura de la señal tomada en el dedo ha sido mejor que en la muñeca. Mediante la incorporación de la doble lectura se ha garantizado la adquisición de una correcta señal de fotopleletismografía, así como también conseguir una mejor repetibilidad y reproducibilidad en las medidas bajo condiciones externas que no sean del todo favorables (p.68).

Así mismo en un contexto continental se hicieron investigaciones importantes con respecto al control del ritmo cardiaco utilizando distintas tecnologías. En el Instituto Politécnico Nacional de México se desarrollaron 2 investigaciones de interés en el tema presente a investigar en el cual el primero que lleva por título “Prototipo de un sistema embebido para el monitoreo del ritmo cardiaco”, perteneciente a Miguel Alejandro solano Bartolo y José Samuel Torres Rodríguez.

(Solano, Torres.2015) concluyeron que el sistema embebido con base en el soft-core MicroBlaze, permite la recepción de los datos y el procesamiento de la información para estimar el ritmo cardiaco, el diseño de una aplicación web basado en PHP, HTML, JavaScript y MYSQL permitieron el correcto manejo de la información recaudada y enviada al sistema embebido.

La otra investigación desarrollada en esta universidad mencionada tiene gran impacto para el presente proyecto y lleva por título “Monitoreo de ritmo cardiaco para dispositivos Android mediante un enlace Bluetooth” perteneciente a Pedro Enrique Velázquez Elizondo y Luis Javier Villagrán Valencia.

(Velázquez, Villagrán.2016) lograron llegar a las siguientes conclusiones:

Se ha demostrado que gracias a las tecnologías actuales disponibles en México se pueden generar herramientas sofisticadas y fáciles de usar para cualquier usuario, las cuales ayudan a la sociedad a ser más consiente y responsable con si salud de una forma cómoda. Esto

porque se genera un sistema el cual puede ser ampliamente usado, ya sea por doctores, deportistas o cualquier persona que requiera del monitoreo de su ritmo cardiaco.

Y por último en la Escuela Politécnica Nacional de Ecuador, se Diseñó un sistema inalámbrico para monitoreo de pacientes ambulatorios, utilizando sensores de presión arterial y ritmo cardiaco e implementación de un prototipo de prueba, perteneciente a Fernanda Tatiana Ortiz Lima.

Después de desarrollar la investigación (Ortiz, 2016) concluye que el diseño e implementación de infraestructuras tecnológicas e telemedicina permite la creación de nuevos servicios en la red enfocados en la movilidad del paciente, al monitoreo de sus signos vitales en tiempo real y su respectivo almacenamiento en la nube. El uso de distintos sensores para la medicina son importantes por que ayuda a prevenir la morbilidad a razón de que se estaría monitoreando al paciente en tiempo real. A su vez, esto permite ahorro en el tiempo, espacio y costo para ambas partes, tanto como para el paciente y para el centro médico.

## 2.2. Marco conceptual

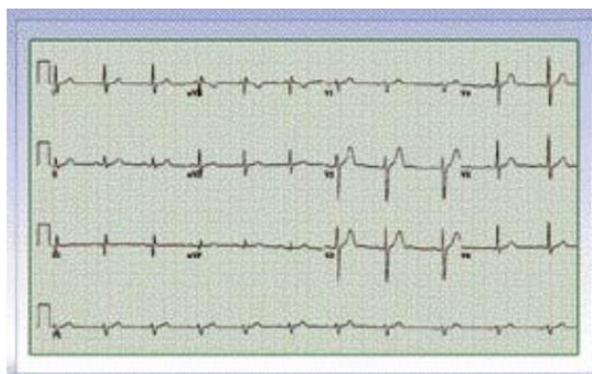
### 2.2.1. Ritmo cardiaco.

El ritmo del pulso es la medida de la frecuencia cardíaca, es decir, del número de veces que el corazón late por minuto. Cuando el corazón impulsa la sangre a través de las arterias, éstas se expanden y se contraen con el flujo de la sangre. Al tomar el pulso no sólo se mide la frecuencia cardíaca, sino que también puede indicar:

El ritmo del corazón.

La fuerza de los latidos.

El pulso normal de los adultos sanos oscila entre 60 y 100 latidos por minuto. El pulso podría fluctuar y aumentar con el ejercicio, las enfermedades, las lesiones y las emociones. Las niñas a partir de los 12 años y las mujeres en general suelen tener el pulso más rápido que los niños y los hombres. Los deportistas, como los corredores, que hacen mucho ejercicio cardiovascular, pueden tener ritmos cardiacos de hasta 40 latidos por minuto sin tener ningún problema(Dalcame,2005).



*Figura 1. Ritmo cardiaco normal*

*Fuente: Electrocardiografía y arritmia*

### 2.2.1.1. Componentes del ritmo cardiaco.

Existen distintos componentes que describen en el libro Electrocardiografía y arritmias de los cuales podemos resaltar los siguientes:

- Onda P que precede a cada complejo QRS.
- Onda P sin la variación de la morfología en una misma derivación.
- Intervalo PR constante.
- Intervalo PP constante con intervalo RR constante.
- Onda P positiva en DI-DII y a VF (Eje inferior por despolarización de arriba-abajo).

#### 2.2.1.1.1. Complejo QRS.

El complejo QRS indica la despolarización ventricular. La despolarización provoca la contracción de los ventrículos.

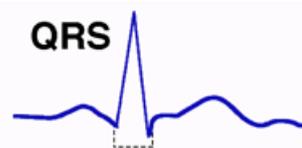


Figura 2. Intervalo QRS.

Fuente: [www.my-ekg.com/generalidades-ekg/intervalos-segmentos-ekg.html](http://www.my-ekg.com/generalidades-ekg/intervalos-segmentos-ekg.html)

#### 2.2.1.1.2. Onda P.

La onda p representa la despolarización auricular, y este a su vez es la despolarización de derecha a izquierda de las aurículas.

#### 2.2.1.1.3. Intervalo RR.

Es la distancia entre dos ondas R sucesivas.



Figura 3. Intervalo RR.

Fuente: [www.my-ekg.com/generalidades-ekg/intervalos-segmentos-ekg.html](http://www.my-ekg.com/generalidades-ekg/intervalos-segmentos-ekg.html)

#### 2.2.1.1.4. Intervalo PR.

Representa la despolarización auricular y el retraso fisiológico que sufre el estímulo a su paso por el nodo Auriculoventricular (AV).

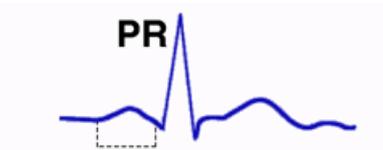


Figura 4. Intervalo PR.

Fuente: [www.my-ekg.com/generalidades-ekg/intervalos-segmentos-ekg.html](http://www.my-ekg.com/generalidades-ekg/intervalos-segmentos-ekg.html)

Al momento de plasmar un electrocardiograma del ritmo cardiaco normal obtenemos una imagen así (Uribe, Duque, & Medina, 2008).

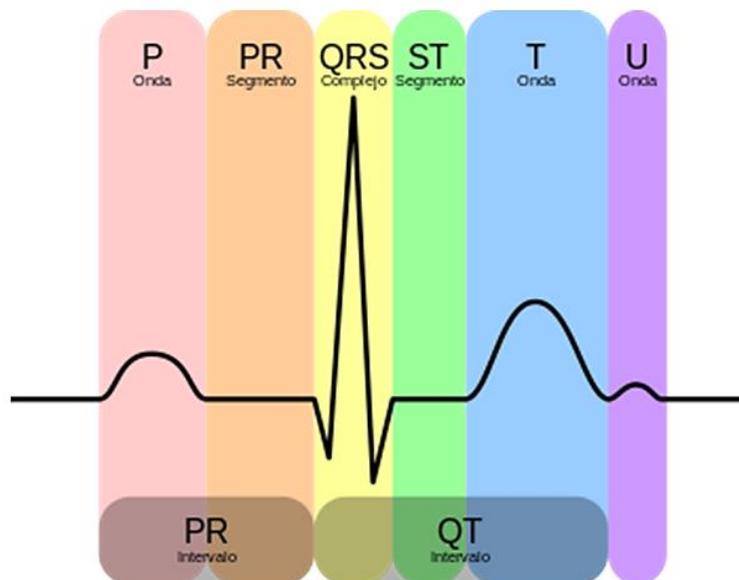


Figura 5. Electrocardiograma.

Fuente: [medicinageneraluniversal.blogspot.com/2017/01/que-es-un-electrocardiograma.html](http://medicinageneraluniversal.blogspot.com/2017/01/que-es-un-electrocardiograma.html)

#### 2.2.2. Internet de las cosas(IOT).

La creación de internet marca una gran trascendencia para la comunicación entre las personas y hoy en día no solo eso también la comunicación e interconexión de los distintos dispositivos que son cotidianos en nuestras vidas.

IOT es La interconexión de dispositivos de detección y activación proporciona la capacidad de compartir información a través de plataformas a través de un marco

unificado, desarrollando una imagen operativa común para permitir aplicaciones innovadoras. Esto se logra mediante la detección ubicua sin fisuras, el análisis de datos y la representación de la información con la computación en la nube como marco unificador(Gubbi, Buyya, Marusic, & Palaniswami, 2013).

### 2.2.3. Gsm.

Es un estándar internacional para teléfonos móviles. Es un acrónimo que significa Sistema Global para Comunicaciones Móviles. También se lo denomina a veces 2G, ya que es una red celular de segunda generación.

Para usar GPRS para el acceso a Internet, y para que el Arduino solicite o brinde servicios a páginas web, debe obtener el nombre del punto de acceso (APN) y un nombre de usuario / contraseña del operador de red. Consulte la información en Conexión a Internet para obtener más información sobre el uso de las capacidades de datos del escudo.

Entre otras cosas, GSM admite llamadas de voz salientes y entrantes, sistema simple de mensajes (SMS o mensajes de texto) y comunicación de datos (a través de GPRS) (Arduino,2018).



Figura 6. Arduino GSM Shield.

Fuente: <https://es.aliexpress.com/item/MINI-V3-8-2-Wireless-Data-Transmission-Extension-Module-GSM-GPRS-SIM900A/32640043214.html>

### 2.2.4. Sensor

Un sensor es conocido como una entrada que tiene la capacidad de captar acciones y estímulos externos ya sean físicos o químicos para convertirlos en eléctricos y su posterior procesamiento según se requiera de este, hoy en día existen distintos tipos de sensores que puede medir desde temperatura, nivel de ph, presión atmosférica hasta ritmo cardiaco, geolocalización, etc. Los sensores pueden emitir salidas

analógicas como digitales a su vez este utilizarse con fines de investigación, innovación, tecnología usable o visible, etc.

#### 2.2.4.1. **Sensor Cardíaco.**

Hoy en día existen distintos tipos de sensores cardiacos entre los cuales podemos destacar:



Figura 7. Variedad de sensores cardiacos.

Fuente: <https://elultimotriatleta.com/mejores-monitores-frecuencia-cardiaca/>

#### 2.2.4.2. **Sensor de pulso cardiaco(SEN-11574).**

Es un sensor de frecuencia cardíaca plug-and-play para Arduino, Teensy y otros microcontroladores. Puede ser utilizado por estudiantes, artistas, atletas, fabricantes y desarrolladores de juegos y dispositivos móviles que deseen incorporar fácilmente datos de frecuencia cardíaca en vivo en sus proyectos. Combina esencialmente un sensor de frecuencia cardíaca óptico simple con circuitos de cancelación de ruido y amplificación que lo hacen rápido y fácil de obtener lecturas de pulso confiables. Además, consume energía con solo 4mA de consumo de corriente a 5V, por lo que es ideal para aplicaciones móviles.



Figura 8. Sensor de pulso cardiaco.

Fuente: [sanderobotics.com/producto/hr0214-37/](https://sanderobotics.com/producto/hr0214-37/)

## 2.2.5. Arduino.

“Arduino es una plataforma de prototipos electrónica de código abierto (open-source) basada

en hardware y software flexibles y fáciles de usar. Está pensado para artistas, diseñadores, como hobby y para cualquiera interesado en crear objetos o entornos interactivos” (Enriquez, 2009.p8).

La placa arduino tiene la capacidad de poder capacidad de sentir el entorno mediante la entrada de una variedad de sensores y a su vez mandar datos después de procesamiento respectivo, es una placa reprogramable y es ahí el gran potencial que se tiene esta máquina.

### 2.2.5.1. *Arduino nano.*

El Arduino Nano es un tablero pequeño, completo y fácil de usar basado en el ATmega328P (Arduino Nano 3.x). Tiene más o menos la misma funcionalidad del Arduino Duemilanove, pero en un paquete diferente. Le falta solo un conector de alimentación de CC, y funciona con un cable USB Mini-B en lugar de uno estándar (Arduino nano, s.f), a su vez también tiene características técnicas propias de esta placa que se detallaran a continuación:

Tabla 1. Especificaciones técnicas del arduino nano.

<b>Microcontrolador</b>	<b>ATmega328</b>
<b>Arquitectura</b>	AVR
<b>Tensión de funcionamiento</b>	5V
<b>Memoria Flash</b>	32 KB de los cuales 2KB utilizado por el gestor de arranque
<b>SRAM</b>	2 KB
<b>Velocidad del reloj</b>	16 MHz
<b>Clavijas Analógicas IN</b>	8
<b>EEPROM</b>	1 KB
<b>Corriente continua por pines de E/S</b>	40 mA(pines de E/S)
<b>Voltaje de entrada</b>	7-12 V
<b>Pines de E/S digitales</b>	22(6 de los cuales son PWM)
<b>Salida PWM</b>	6
<b>El consumo de energía</b>	19 mA

<b>Tamaño del PBC</b>	18 x 45mm
<b>Peso</b>	7 g

*Fuente: [store.arduino.cc/usa/arduino-nano](http://store.arduino.cc/usa/arduino-nano).*

### 2.2.6. Servidor web.

Se puede describir como servidor a toda aquella máquina, computadora, o aparato electrónico que está al servicio de otros dispositivos que pueden ser: computadoras, celulares, impresoras, etc.

En la informática existen distintos tipos de servidores, los cuales detallaremos a continuación.

*Tabla 2. Tipos de servidor.*

DENOMINACIÓN DEL SERVIDOR	DESCRIPCIÓN
<b>Servidor de Correo</b>	Es el servidor que almacena, envía, recibe y realiza todas las operaciones relacionadas con el e-mail de sus clientes.
<b>Servidor Proxy</b>	Es el servidor que actúa de intermediario de forma que el servidor que recibe una petición no conoce quién es el cliente que verdaderamente está detrás de esa petición.
<b>Servidor Web</b>	Almacena principalmente documentos HTML (son documentos a modo de archivos con un formato especial para la visualización de páginas web en los navegadores de los clientes), imágenes, videos, texto, presentaciones, y en general todo tipo de información. Además se encarga de enviar estas informaciones a los clientes.

<p><b>Servidor de Base de Datos</b></p>	<p>Da servicios de almacenamiento y gestión de bases de datos a sus clientes. Una base de datos es un sistema que nos permite almacenar grandes cantidades de información. Por ejemplo, todos los datos de los clientes de un banco y sus movimientos en las cuentas.</p>
<p><b>Servidores Clúster</b></p>	<p>Son servidores especializados en el almacenamiento de la información teniendo grandes capacidades de almacenamiento y permitiendo evitar la pérdida de la información por problemas en otros servidores.</p>
<p><b>Servidores Dedicados</b></p>	<p>Como ya expresamos anteriormente, hay servidores compartidos si hay varias personas o empresas usando un mismo servidor, o dedicados que son exclusivos para una sola persona o empresa.</p>
<p><b>Servidores de imágenes</b></p>	<p>Recientemente también se han popularizado servidores especializados en imágenes, permitiendo alojar gran cantidad de imágenes sin consumir recursos de nuestro servidor web en almacenamiento o para almacenar fotografías personales, profesionales, etc. Algunos gratuitos pueden ser: <a href="http://www.imgur.com">www.imgur.com</a>, <a href="http://www.photobucket.com">www.photobucket.com</a>, <a href="http://www.flickr.com">www.flickr.com</a> de Yahoo, o <a href="http://picasaweb.google.com">picasaweb.google.com</a> de Google.</p>

*Fuente: Elaboración propia.*

### **2.2.7. Bootstrap.**

Bootstrap es un conjunto de herramientas de código abierto para desarrollar con HTML, CSS y JS. Realice rápidamente un prototipo de sus ideas o construya su aplicación completa con nuestras variables y combinaciones Sass, sistema de

cuadrícula sensible, componentes precompilados extensos y potentes complementos integrados en jQuery. (@mdo & @fat, n.d.).

#### **2.2.8. ThingSpeak.**

ThingSpeak es un servicio de plataforma de análisis de IoT que le permite agregar, visualizar y analizar flujos de datos en vivo en la nube. ThingSpeak proporciona visualizaciones instantáneas de los datos publicados por sus dispositivos en ThingSpeak. Con la capacidad de ejecutar el código MATLAB® en ThingSpeak, puede realizar el análisis y el procesamiento en línea de los datos a medida que se incorporan. ThingSpeak se usa a menudo para la creación de prototipos y los sistemas de prueba de concepto IoT que requieren análisis (“Learn More - ThingSpeak IoT,” n.d.).

#### **2.2.9. Tecnología usable(wearables).**

La tecnología que se va desarrollando cada vez más y es así que los wearables o también llamada tecnología usable están dando de qué hablar hoy en día, pues se trata de la tecnología que fácilmente lo podemos usar como cualquier indumentaria que llevamos puestos, no es más que otra prenda para las personas que tiene gran utilidad para poder determinar distintas actividades de nuestro cuerpo humano como es la temperatura, distancia que recorremos, cantidad de pasos que damos, nuestra presión arterial, ritmo cardiaco y hasta incluso se usa por placer en prendas de vestir que adquieren peculiaridades como cambiar de colores y demás propiedades que se puede realizar con aparatos electrónicos. Además, el objetivo que tiene los wearables es controlar las funciones vitales de las personas, así como interconectarlas a través de este tipo de tecnologías(Park & Jayaraman, 2003).

#### **2.2.10. Metodologías ágiles.**

Al 2018 existen diversos tipos de metodologías para el desarrollo de aplicaciones, software y programas y dependiendo del nivel de complejidad se hace uso de las metodologías, y una de las mejores opciones es el uso de las metodologías ágiles, bueno ¿Qué son las metodologías ágiles? Según manifiestan (Canós, Letelier y Penadés, 2012).

Tras esta reunión se creó The Agile Alliance, una organización, sin ánimo de lucro, dedicada a promover los conceptos relacionados con el desarrollo ágil de software y ayudar a las organizaciones para que adopten dichos conceptos. El punto de partida es fue el Manifiesto Ágil, un documento que resume la filosofía “ágil” (p.02).

El manifiesto ágil está compuesto de la siguiente manera.

##### **2.2.10.1. El Manifiesto Ágil.**

Según el Manifiesto se valora:

**Al individuo y las interacciones del equipo de desarrollo sobre el proceso y las herramientas.** La gente es el principal factor de éxito de un proyecto software. Es más importante construir un buen equipo que construir el entorno. Muchas veces se comete el error de construir primero el entorno y esperar que el equipo se adapte automáticamente. Es mejor crear el equipo y que éste configure su propio entorno de desarrollo en base a sus necesidades.

**Desarrollar software que funciona más que conseguir una buena documentación.** La regla a seguir es “no producir documentos a menos que sean necesarios de forma inmediata para tomar una decisión importante”. Estos documentos deben ser cortos y centrarse en lo fundamental.

**La colaboración con el cliente más que la negociación de un contrato.** Se propone que exista una interacción constante entre el cliente y el equipo de desarrollo. Esta colaboración entre ambos será la que marque la marcha del proyecto y asegure su éxito.

**Responder a los cambios más que seguir estrictamente un plan.** La habilidad de responder a los cambios que puedan surgir al largo del proyecto (cambios en los requisitos, en la tecnología, en el equipo, etc.) determina también el éxito o fracaso del mismo. Por lo tanto, la planificación no debe ser estricta sino flexible y abierta.

Los valores anteriores inspiran los doce principios del manifiesto. Son características que diferencian un proceso ágil de uno tradicional. Los dos primeros principios son generales y resumen gran parte del espíritu ágil. El resto tienen que ver con el proceso a seguir y con el equipo de desarrollo, en cuanto metas a seguir y organización del mismo. Los principios son:

- I. La prioridad es satisfacer al cliente mediante tempranas y continuas entregas de software que le aporte un valor.
- II. Dar la bienvenida a los cambios. Se capturan los cambios para que el cliente tenga una ventaja competitiva.
- III. Entregar frecuentemente software que funcione desde un par de semanas a un par de meses, con el menor intervalo de tiempo posible entre entregas.
- IV. La gente del negocio y los desarrolladores deben trabajar juntos a lo largo del proyecto.

- V. Construir el proyecto en torno a individuos motivados. Darles el entorno y el apoyo que necesitan y confiar en ellos para conseguir finalizar el trabajo.
- VI. El diálogo cara a cara es el método más eficiente y efectivo para comunicar información dentro de un equipo de desarrollo.
- VII. El software que funciona es la medida principal de progreso.
- VIII. Los procesos ágiles promueven un desarrollo sostenible. Los promotores, desarrolladores y usuarios deberían ser capaces de mantener una paz constante.
- IX. La atención continua a la calidad técnica y al buen diseño mejora la agilidad.
- X. La simplicidad es esencial.
- XI. Las mejores arquitecturas, requisitos y diseños surgen de los equipos organizados por sí mismos.
- XII. En intervalos regulares, el equipo reflexiona respecto a cómo llegar a ser más efectivo, y según esto ajusta su comportamiento. (Canós, Letelier y Penadés, 2012).

#### **2.2.10.2. Programación Extrema.**

Esta metodología fue formulada por Kent Beck, autor del primer libro sobre la materia, Extreme Programming Explained: Embrace Change en 1999, esta metodología viene a ser un enfoque en la ingeniería de software que tiene ventajas muy provechosas respecto a las metodologías tradicionales, por qué le pone énfasis en la adaptabilidad que en la previsibilidad.

Principales Características:

- Desarrollo iterativo e incremental: pequeñas mejoras, unas tras otras.
- Pruebas unitarias continuas, frecuentemente repetidas y automatizadas, incluyendo pruebas de regresión. Se aconseja escribir el código de la prueba antes de la codificación. Programación en parejas: se recomienda que las tareas de desarrollo se lleven a cabo por dos personas en un mismo puesto. Se supone que la mayor calidad del código escrito de esta manera el código es revisado y discutido mientras se escribe y es más importante que la posible pérdida de productividad inmediata.

- Frecuente integración del equipo de programación con el cliente o usuario. Se recomienda que un representante del cliente trabaje junto al equipo de desarrollo.
- Corrección de todos los errores antes de añadir nueva funcionalidad. Hacer entregas frecuentes.
- Refactorización del código, es decir reescribir ciertas partes del código para aumentar su legibilidad y mantenibilidad, pero sin modificar su comportamiento. Las pruebas han de garantizar que en la refactorización no se ha introducido ningún fallo.
- Propiedad del código compartida: en vez de dividir la responsabilidad en el desarrollo de cada módulo en grupos de trabajo distintos, este método promueve el que todo el personal pueda corregir y extender cualquier parte del proyecto. Las frecuentes pruebas de regresión garantizan que los posibles errores serán detectados.
- Simplicidad en el código: es la mejor manera de que las cosas funcionen. Cuando todo funcione se podrá añadir funcionalidad si es necesario. La programación extrema apuesta que es más sencillo hacer algo simple y tener un poco de trabajo extra para cambiarlo si se requiere, que realizar algo complicado y quizás nunca utilizarlo (Unad, 2010).

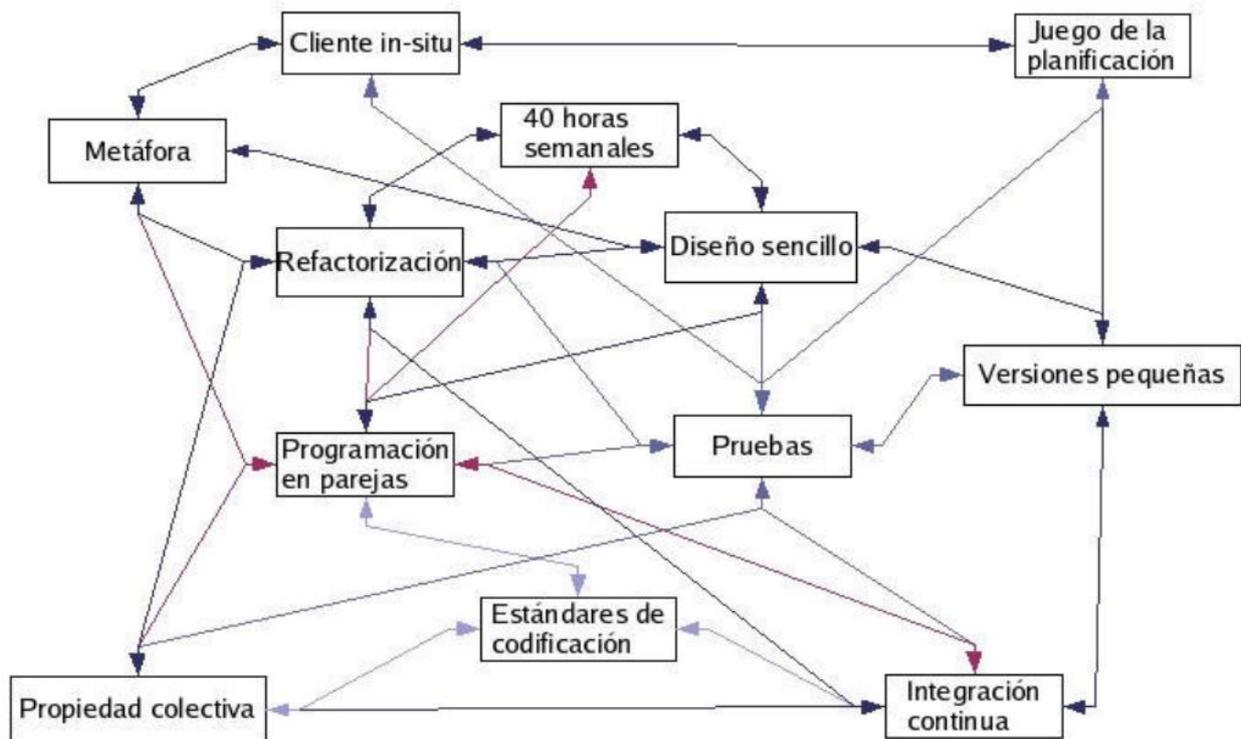


Figura 9. Las prácticas se refuerzan entre sí.

Fuente: [www.cyta.com.ar/ta0502/v5n2a1.htm](http://www.cyta.com.ar/ta0502/v5n2a1.htm)

### Fases de la programación extrema:

#### 2.2.10.2.1. Planificación:

Es la primera actividad en el proceso de desarrollo. Comienza creando una serie de historias de usuarios, similares a los casos de uso que describen la funcionalidad del software que se va a construir. El cliente les asigna una prioridad y el equipo de desarrollo XP evalúa cada una y le asigna un periodo de desarrollo. Si la historia supera más de tres semanas de desarrollo se divide la historia en historias menores. Una vez establecido un acuerdo detallando la fecha de entrega, el equipo de desarrollo ordena las historias para implementar antes las que tengan mayor prioridad. Conforme avanza el trabajo de desarrollo, el cliente puede agregar nuevas historias con nueva funcionalidad, de esta manera, la programación extrema es una metodología que acepta fácilmente requisitos cambiantes durante el desarrollo de software.

2.2.10.2.2. Diseño:

El diseño en la programación extrema sigue el principio de hacerlo todo simple. El diseño se va modificando a lo largo de todo el proceso de desarrollo.

2.2.10.2.3. Desarrollo:

La programación extrema recomienda que después de diseñar las historias el equipo no deba comenzar la codificación, sino que debe desarrollar una serie de pruebas de unidad que les ayuden a centrarse en lo que debe implementarse para pasar esa prueba. Un concepto clave en esta etapa es la programación en pareja de tal forma que dos personas trabajan juntas en un ordenador para crear el código de una historia siguiendo un estándar de codificación. Este enfoque asegura la calidad del código, ya que dos cabezas piensan mejor que una, además permite la rápida comunicación entre las dos personas y un mejor conocimiento del problema que se quiere solucionar.

2.2.10.2.4. Pruebas:

Las pruebas de unidad creadas deben ser automatizadas para que puedan ejecutarse de manera fácil y rápida. De esta forma podemos modificar el código y asegurarnos que funciona pese a los cambios producidos (Villca, 2016, p.42).

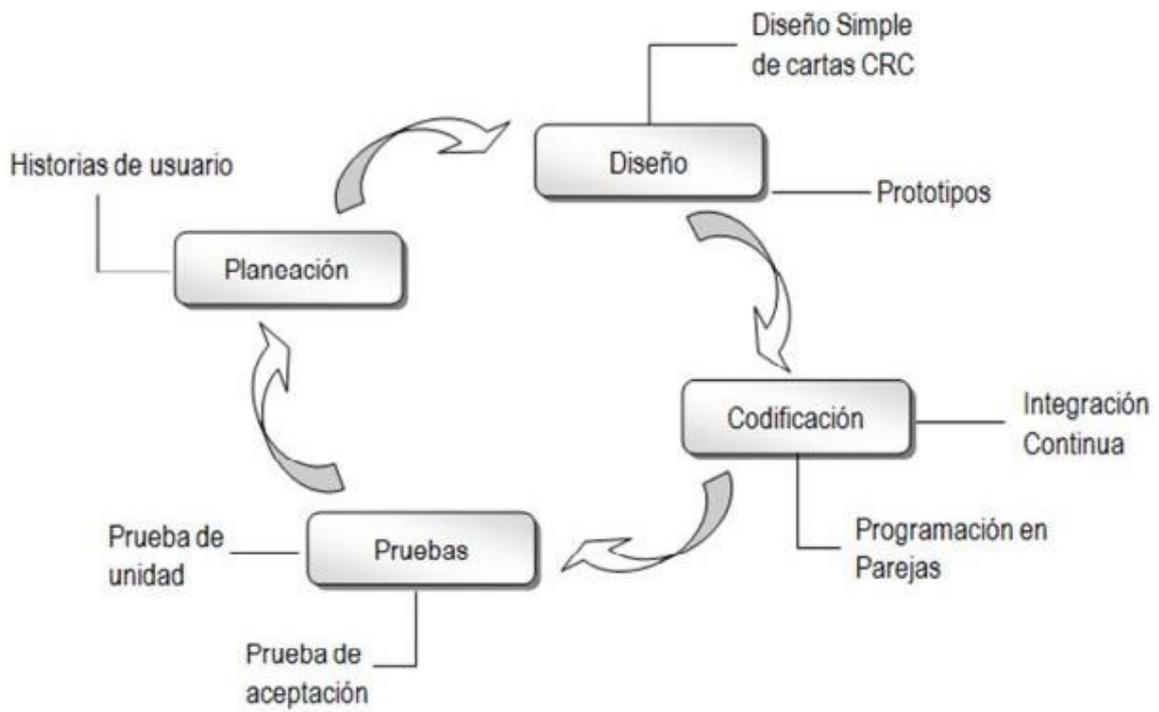


Figura 10. Fases de la Programación Extrema.

Fuente: [isprogramacionextrema.blogspot.com/2013/09/estructura-y-elementos.html](http://isprogramacionextrema.blogspot.com/2013/09/estructura-y-elementos.html).

## CAPÍTULO 3: MATERIALES Y MÉTODOS

En este capítulo se presentará el desarrollo del proyecto utilizando las fases de la Programación Extrema, así como también se describirá el uso de las herramientas y distintos componentes que se utilizó en el presente proyecto.

### 3.1. Metodología para la implementación del proyecto (Programación Extrema)

Para la implementación del presente proyecto se vio por conveniente desarrollarlo basándonos en la metodología de desarrollo ágil Programación Extrema(XP), ya que cumplía con la estructura y técnica adecuada y ciclos iterativos que coadyuvaron a poder desarrollar con plenitud y éxito el presente proyecto ya que tiene.

### 3.2. Herramientas Empleadas

A continuación, se hará de presente las distintas herramientas empleadas, tanto software como hardware.

#### 3.2.1. Software.

##### 3.2.1.1. Visual Studio Code.

Es un editor de código fuente libre y gratuito, fue desarrollado por Microsoft, compatible para los distintos sistemas operativos de ordenadores y lo más resaltante es que tiene soporte para depuración control de Git integrado, resaltado de sintaxis, finalización de código inteligente, fragmentos de código y refactorización de código.

##### 3.2.1.2. Xampp.

Servidor web HTTP multiplataforma con base de datos MariaDB muy utilizado y de preferencia para poder realizar las pruebas con los distintos aparatos.

##### 3.2.1.3. Php.

Lenguaje de programación orientado a web con gran uso en la comunidad de desarrolladores web, su facilidad de comprensión y sintaxis, así como su usabilidad en la Programación Orientada a Objetos y adaptabilidad al patrón arquitectónico Modelo Vista Controlador(MVC).

##### 3.2.1.4. Arduino Ide.

Es el entorno de programación para arduino por el cual se desarrolló las pruebas y distintas codificaciones que ayudaron a ejecutar el proyecto de forma correcta.

### **3.2.2. Hardware.**

En el caso del hardware solo pasare a mencionarlo ya que la descripción detallada se encuentra en el marco teórico.

- Arduino nano.
- Sensor de pulso cardiaco.
- Protoboard.
- SIM900.

## CAPÍTULO 4: RESULTADOS Y DISCUSIÓN

### 4.1. Implementación de la metodología ágil Programación Extrema(XP)

#### 4.1.1. Planificación.

A partir del presente se describe la experiencia obtenida al momento de la ejecución del proyecto utilizando la metodología XP.

Iniciaremos describiendo esta primera fase que toma XP para el desarrollo del proyecto, haciendo uso de las distintas herramientas que nos proporciona XP, así como esta contrastado en el marco teórico entre los cuales se encuentra las historias de usuarios el plan de entrega.

##### 4.1.1.1. Historia de Usuario.

El cliente es quien escribe a través de esta herramienta de la metodología XP para poder expresar los requerimientos que son necesarios para la funcionalidad del proyecto, lo más resaltante de esta parte es que el usuario puede utilizar su lenguaje natural sin necesidad de tener conocimiento sobre aspectos técnicos para el desarrollo del proyecto.

También se logró extraer satisfactoriamente la información necesaria para poder desarrollar correctamente sin demasiadas aclaraciones por parte del cliente.

Lo fundamental en el uso de la historia de usuario es que existe una comprensión más orgánica y se limita al retraso, por falta de información, ya que se escribe con claridad los requerimientos para el proyecto y a su vez fueron de gran utilidad para la estimación de los tiempos requeridos para el proyecto.

Se consideró 8 historias de usuario para el proyecto ejecutado.

<b>Historia de Usuario</b>	
<b>Numero:</b>	Nombre Historia Usuario:
<b>Usuario:</b>	
<b>Modificación de Historia N°:</b>	Iteración Asignada:
<b>Prioridad en Negocio: (Alta/Media/Baja)</b>	Puntos Estimados:
<b>Riesgo en Desarrollo: (Alta/Media/Baja)</b>	Puntos Reales:
<b>Descripción:</b>	
<b>Observaciones:</b>	

*Figura 11. Historia de Usuario.*

*Fuente: Elaboración propia.*

#### **4.1.1.2. Plan de publicaciones (Relese Planning).**

Se desarrolló una primera reunión con el usuario en el cual se especificaron las primeras instancias y requerimientos del proyecto, en una segunda reunión se validaron las historias de usuario, concertando los grados de importancia y dando las modificaciones necesarias para iniciar con la siguiente fase de la metodología, gracias a la claridad que se mostró con el uso de las herramientas de la metodología no fueron necesarias más reuniones para esta primera fase.

Aunque XP propone que el cliente sea quien decida cuales historias se implementarán y cuál es su grado de importancia de cada una de las iteraciones, pero se optó por la mejor forma de trabajo así que se decidió en conjunto incluyendo al equipo de trabajo y el usuario. Lo cual no genero problemas en las entregas de los módulos funcionales.

Tal y como lo especifica XP la clasificación se realizó según a su orden de importancia en el proyecto.

Y por último se consideró tres iteraciones dentro del proyecto, y para determinar el tiempo se tomó como medida 3 semanas y cada semana estuvo compuesta por 4 días que son lunes, martes, miércoles y jueves,

y cada día se consideró trabajar 5 horas y esto se aprobó por el equipo de ejecución, por las circunstancias de la ejecución del proyecto y otros factores fisiológicos.

#### 4.1.1.2.1. Primera Iteración.

Para esta primera iteración se consideró 3 historias de usuario las cuales son:

- Control de ingreso al sistema.
- Funcionalidad del INTCAP.
- Envío del ritmo cardiaco.



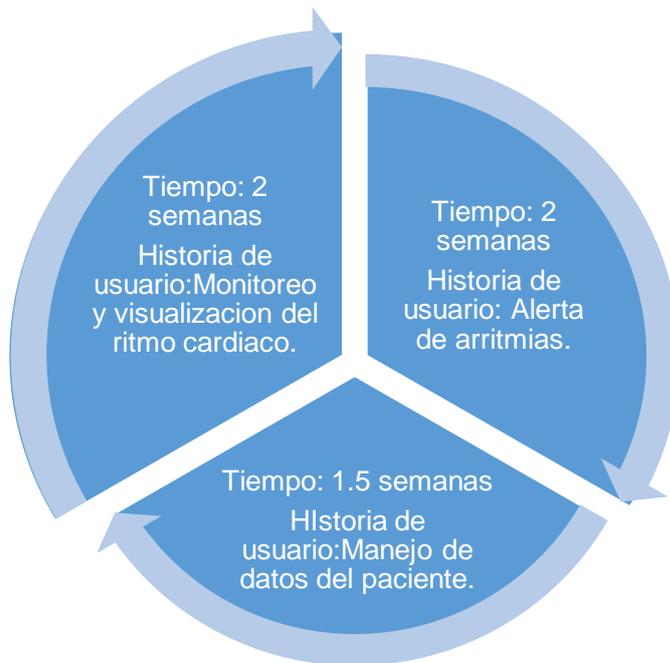
*Gráfico 1. Primera iteración.*

*Fuente: Elaboración propia.*

#### 4.1.1.2.2. Segunda Iteración.

En esta segunda iteración se consideró 3 historias de usuario las cuales son:

- Facilidad en el uso.
- Monitoreo y visualización del ritmo cardiaco.
- Manejo de datos del cliente.



*Gráfico 2. Segunda iteración.*

*Fuente: Elaboración propia.*

#### 4.1.1.2.3. Tercera Iteración.

A diferencia de los casos anteriores en esta iteración se consideraron 2 historias de usuarios y los aspectos de estándares para el sistema que requirió el usuario.

- Alerta de problemas cardiacos.
- Visualización en distintos dispositivos.



Gráfico 3. Tercera iteración.

Fuente: Elaboración propia.

**4.1.1.3. Velocidad del proyecto.**

Se midió la cantidad de tiempo estimado que se requiere para la ejecución de cada iteración en función de las historias de usuario, el nivel de prioridad que estas contemplan y la complejidad que requirió su desarrollo, hubo pequeñas variaciones en el tiempo que se requirió para su completo desarrollo ya que las complejidades de algunas historias fueron descubiertas al momento de su desarrollo.

Tabla 3. Cuadro de iteraciones.

	Iteración 1	Iteración 2	Iteración 3
<b>Horas</b>	<b>60</b>	<b>110</b>	<b>50</b>
<b>Semanas</b>	<b>3</b>	<b>5.5</b>	<b>2.5</b>
<b>Horas semanales</b>	<b>20</b>	<b>20</b>	<b>20</b>
<b>Historias de usuarios</b>	<b>3</b>	<b>3</b>	<b>2</b>

Fuente: Elaboración propia.

#### 4.1.1.4. **Entregas pequeñas.**

Aquí se desarrolló las pequeñas entregas funcionales según el desarrollo de las distintas historias de usuarios para su verificación por el usuario, esto fue fundamental para el éxito por su participación del usuario y al ver el avance del sistema su interés fue mayor para el desarrollo exitoso del proyecto.

Para las entregas se fijaron las siguientes fechas:

*Tabla 4. Cuadro de entregas.*

	Fecha de entrega	Tiempo de la reunión
<b>Primera iteración</b>	20 de agosto	45 minutos
<b>Segunda iteración</b>	2 de octubre	60 minutos
<b>Tercera iteración</b>	22 octubre	90 minutos

*Fuente: Elaboración propia.*

#### 4.1.1.5. **Programación en parejas.**

Tal y como se especifica y lo recomienda XP la programación debe realizarse en parejas para que puedan cerciorarse de la funcionalidad del código de forma bilateral, al inicio se consideró un solo ejecutor, pero dadas las circunstancias de la complejidad y para el éxito del proyecto, se incorporó un personal que coadyuve y colabore para llevar de la mejor forma el presente proyecto.

#### 4.1.1.6. **Reuniones diarias.**

Así como está previsto en la XP, antes de desarrollar diariamente se tomó un tiempo prudente para poder analizar los problemas detectados y las estrategias para poder solucionarlos, así también hablar del itinerario del día y las metas según las fechas estipuladas en el presente proyecto.

La utilización del internet fue fundamental ya que las reuniones que no se podían desarrollarse en presencia física gracias a esta tecnología nos permitió desarrollar este aspecto dentro de XP a cabalidad, así también se utilizaron otras tecnologías como las llamadas telefónicas para consultas y consulta de avances.

#### **4.1.2. Diseño.**

Esta mitología se diferencia de otras por que el diseño se realiza durante el tiempo en el que se desarrolla el proyecto, su revisión es constante y sus modificaciones desarrolladas debido a los posibles cambios existentes.

Uno de los elementos más importantes en XP son las tarjetas CRC, que nos da gran soporte para el desarrollo exitoso del proyecto.

##### **4.1.2.1. Diseños simples.**

Según manifiesta XP, el diseño debe de desarrollarse de forma simple para su facilidad en la implementación así que en esta etapa se elaboró lo necesario y fundamental para el proyecto. Primeramente, se elaboró el diagrama de flujo de datos y la elaboración de la interfaz gráfica, que a su vez estas dos herramientas ayudaron a la fácil comprensión al cliente; existen múltiples herramientas, pero se tomaron las más necesarias para el correcto desarrollo del proyecto.

Cabe recalcar que los usos de estas dos herramientas fueron suficientes para su correcto desarrollo del presente proyecto, además el cliente estuvo conforme cuando se le mostro los avances en las respectivas iteraciones cuando se sostuvieron reuniones. A continuación, se mostrará los diagramas elaborados y las interfaces diseñadas.

Primeramente, se presentará los diagramas de flujo de datos.

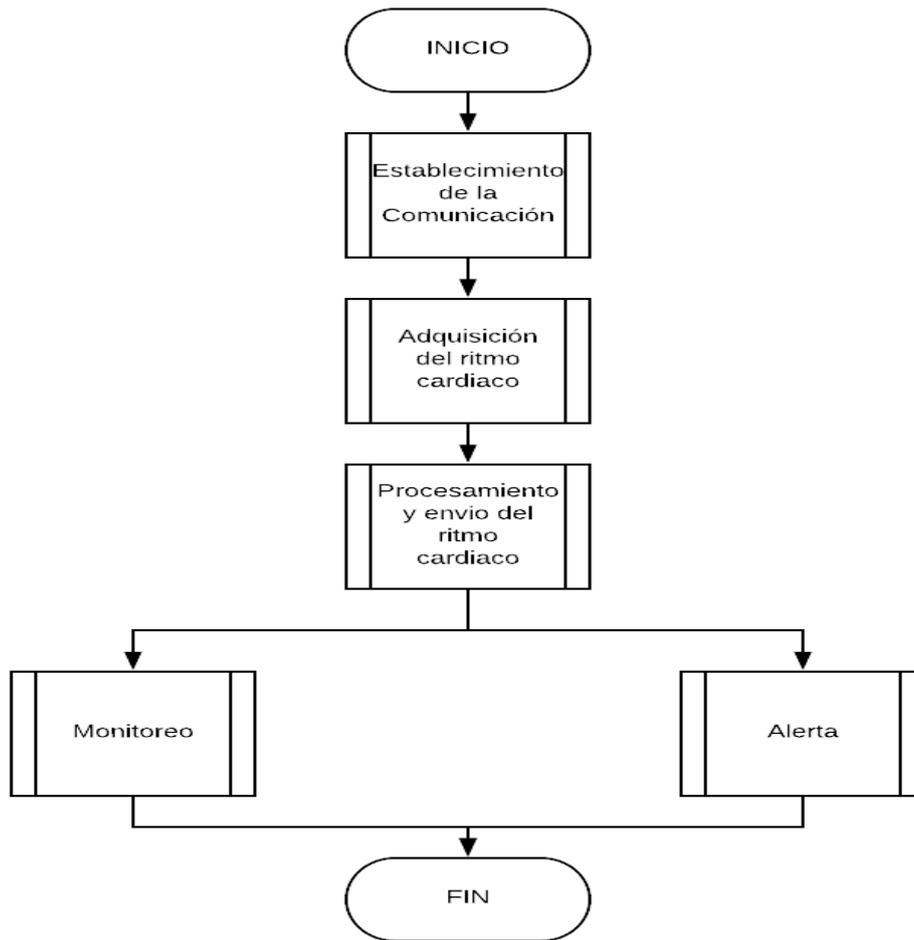


Figura 12. Diagrama de flujo de datos principal.

Fuente: Elaboración propia

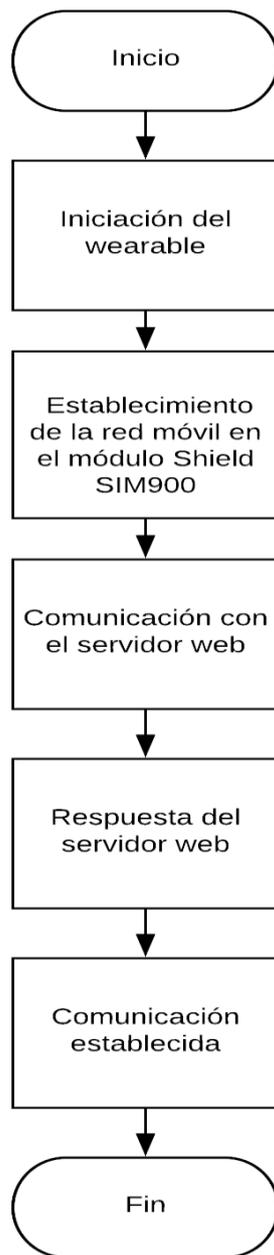
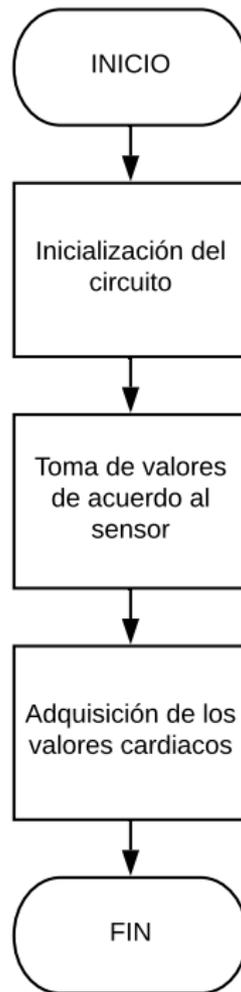


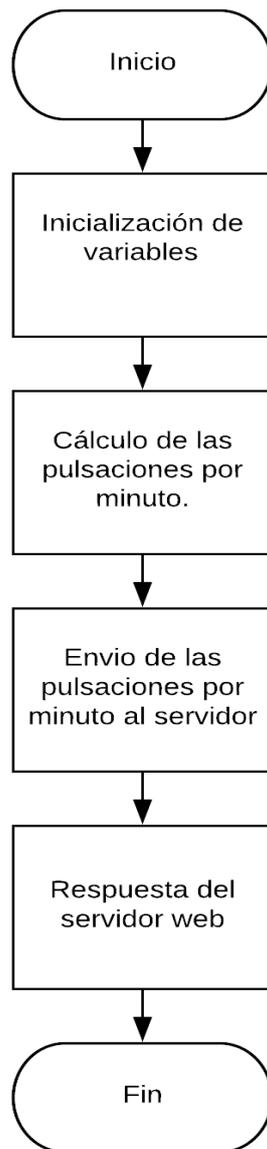
Figura 13. Diagrama de flujo de datos: Establecimiento de la comunicación.

Fuente: Elaboración propia.



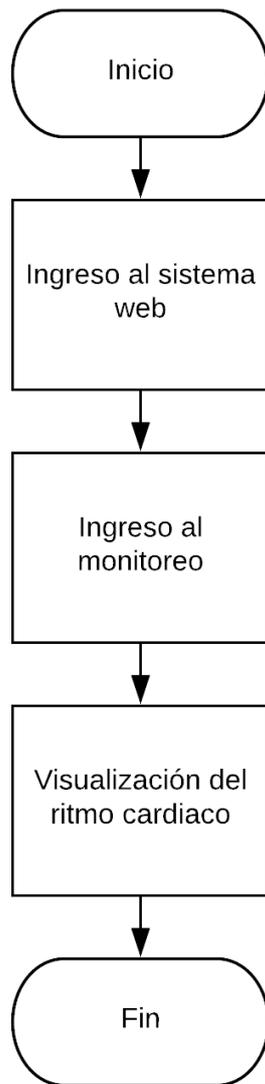
*Figura 14. Diagrama de flujo de datos: Adquisición del ritmo cardiaco.*

*Fuente: Elaboración propia.*



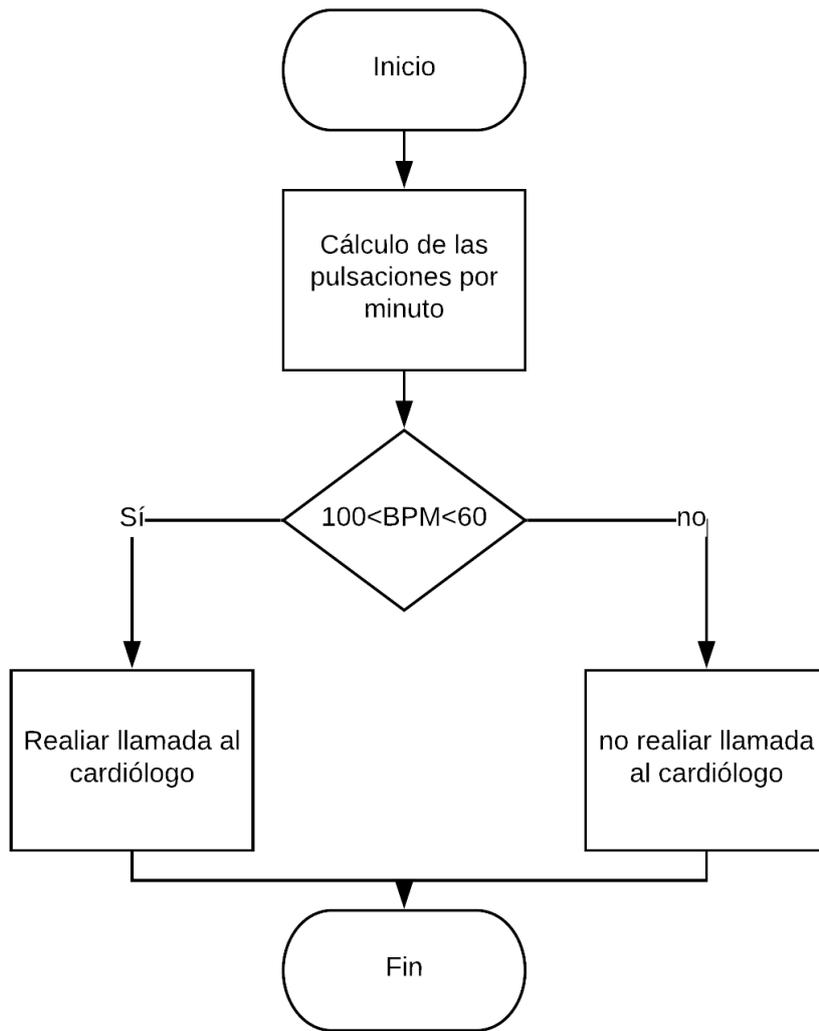
*Figura 15. Diagrama de flujo de datos: Procesamiento y envío del ritmo cardiaco.*

*Fuente: Elaboración propia.*



*Figura 16. Diagrama de flujo de datos: Monitoreo.*

*Fuente: Elaboración propia.*



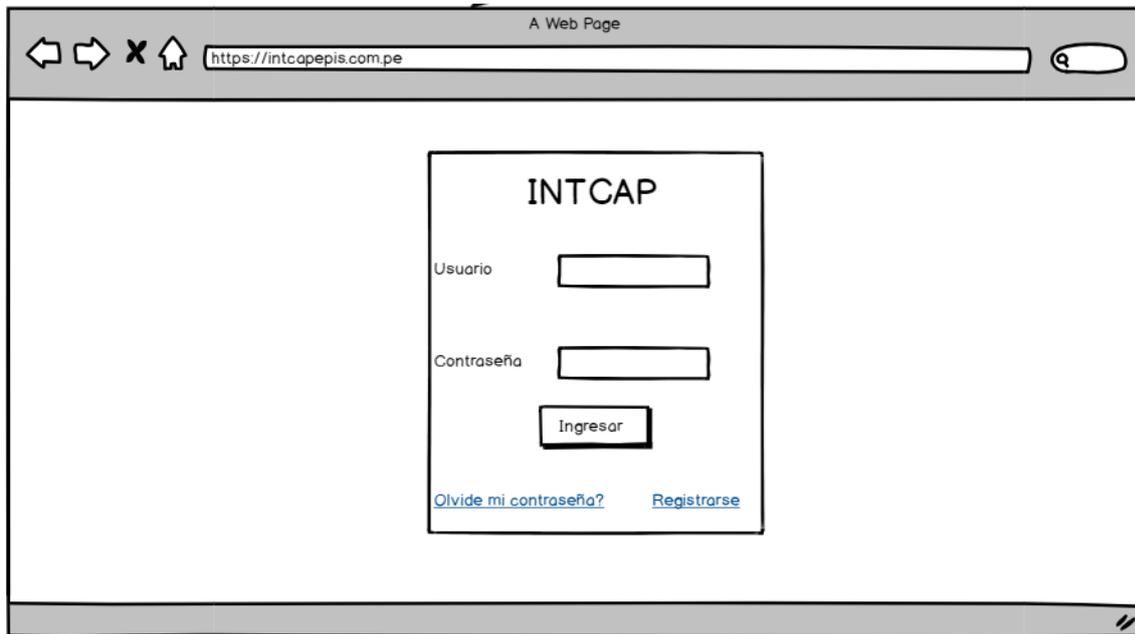
*Figura 17. Diagrama de flujo de datos: Alerta.*

*Fuente: Elaboración propia.*

Ahora se mostrará la maquetación de la página web para su posterior implementación, para la cual se utilizó una herramienta web que por nombre lleva balsamiq.cloud que ayuda a la maquetación de este tipo de sistemas orientados a la web.

En primera instancia se tiene la interfaz de control de ingreso al sistema, esto desarrolló a razón de que se cumpla con aspectos fundamentales de seguridad en manejo de la información.

Seguidamente se presentará la interfaz principal del sistema web, perteneciente a la historia de usuario que hace referencia a la funcionalidad del sistema.



*Figura 18. Maquetación interfaz de control de ingreso al sistema.*

*Fuente: Elaboración propia.*

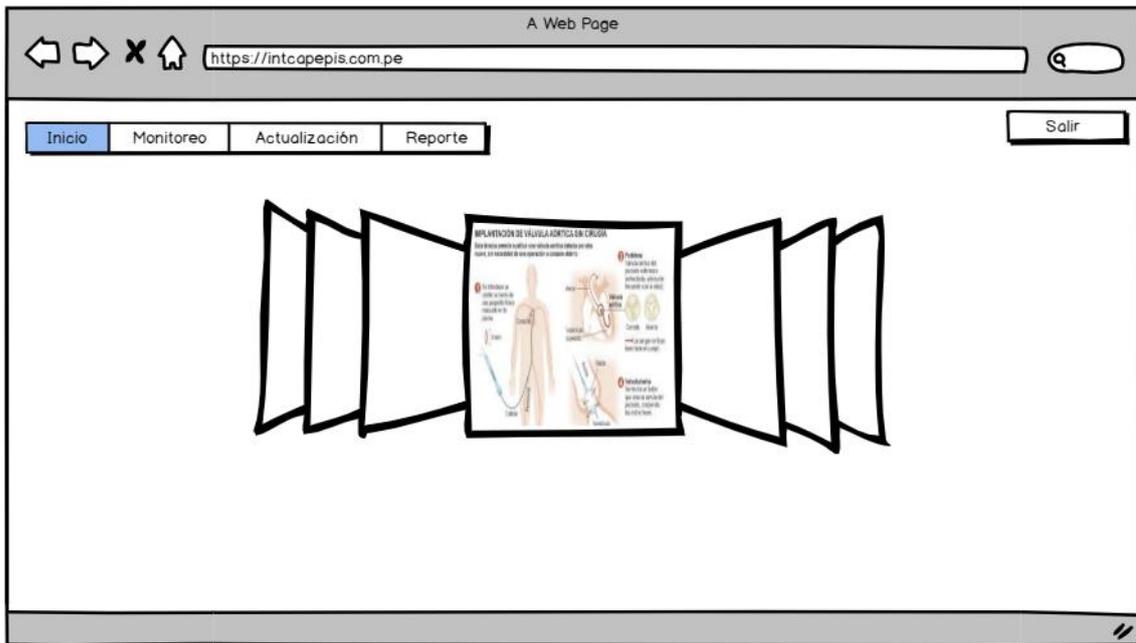


Figura 19. Maquetación interfaz de ventana principal.

Fuente: Elaboración propia.

Ahora se mostrará cada una de las interfaces concernientes a los menús estipulados en la interfaz principal que en conjunto cumplen con la historia de usuario funcionalidad del sistema.



Figura 20. Maquetación menú monitoreo.

Fuente: Elaboración propia.

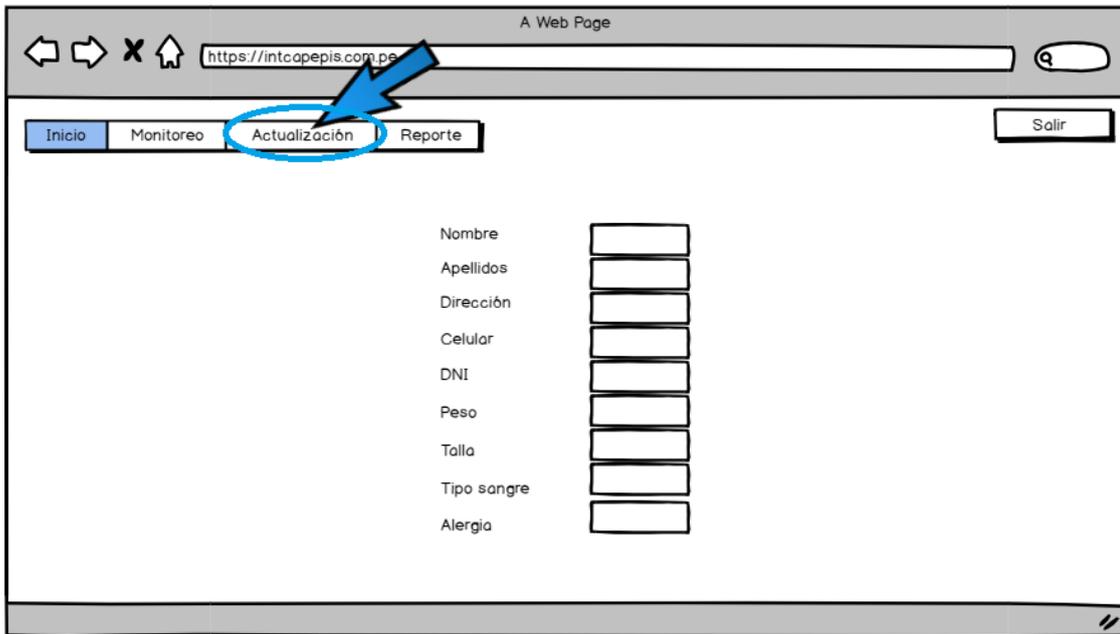


Figura 22. Maquetación menú actualización.

Fuente: Elaboración propia.

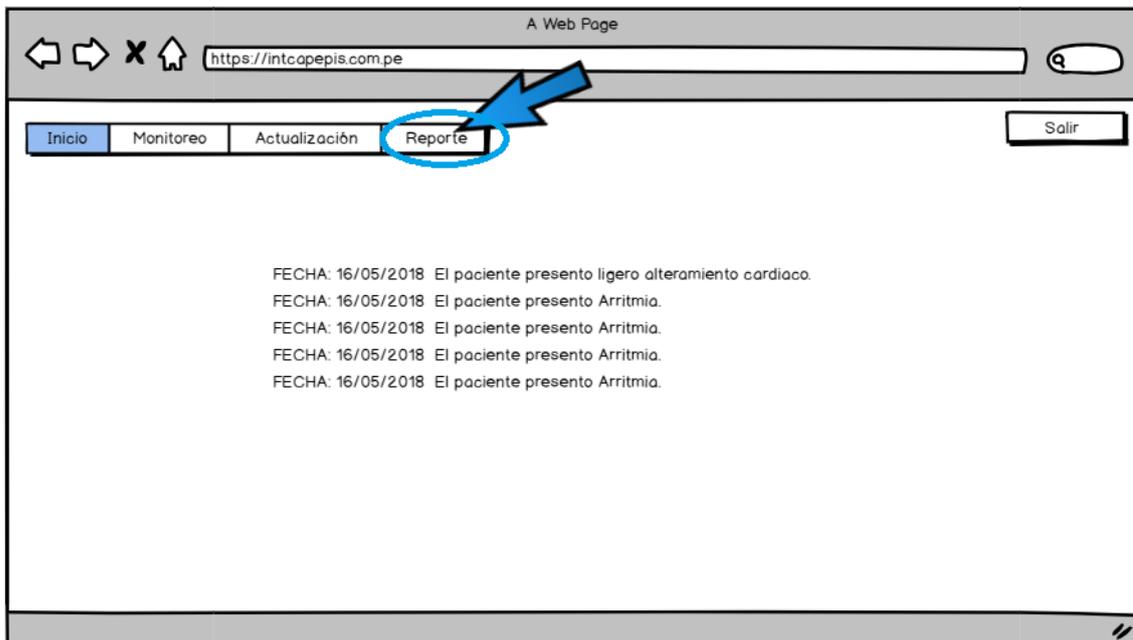


Figura 21. Maquetación menú reporte.

Fuente: Elaboración propia.

**4.1.2.2. Glosario de términos.**

La implementación de este punto no fue necesario a razón de que es una investigación de pregrado y según la recomendación de XP, el desarrollo del diseño debe desarrollarse de con la mejor simplicidad para su fácil comprensión, as también cabe mencionar que otra de las motivaciones fue la carencia de tiempo para poder desarrollarlo con amplitud este punto.

Este punto no influyo en el correcto desarrollo del proyecto y también es preciso mencionar que el código está debidamente comentado como una buena práctica de programación.

**4.1.2.3. Riesgos.**

En este aspecto XP recomienda en el caso que existiera un riesgo que tenga significancia en el desarrollo correcto del proyecto en el diseño darle una atención especial y asignar una pareja de programadores para que puedan solucionarlo, pero en el caso del presente proyecto no existió riesgo considerable alguno, así es que en este punto no se tubo relevancia mayor alguna.

**4.1.2.4. Funcionalidad extra.**

XP recomienda no agregar funcionalidad que no esté estipulado por el cliente, pero en esta ocasión se agregó funcionalidad extra que es la de registro de nuevo usuario para el sistema, que se presentara a continuación.

A Web Page

https://intcapepis.com.pe

Nombre

Apellidos

DNI

E-mail

Dirección

Celular

Contraseña

*Fuente: Elaboración propia.*

*Figura 23. Maquetación menú reporte.*

#### **4.1.2.5. Refactorizar.**

Para la refactorización XP nos indica que se revisa para mejorar y modificar código con el objetivo de optimizar su rendimiento, para el presente proyecto se refactorizó el código de la placa arduino nano, para mejorar el código y no generar código obsoleto.

#### **4.1.2.6. Tarjetas CRC.**

Tal como lo expuesto por XP las tarjetas CRC nos colaboran para la concentración y el desarrollo óptimo del código fuente orientado a objetos.

A continuación, se le presenta las distintas tarjetas CRC que se utilizó.

Tabla 5. Tarjeta CRC: Clase usuario.

<b>Cardiólogo</b>	
Monitorear paciente.	Servidor web, ThingSpeak.
Atender alerta.	Librerías PHP.
Registrarse	Wearable.

Fuente: Elaboración propia.

Tabla 6. Tarjeta CRC: Clase paciente.

<b>Paciente</b>	
Actualización de datos.	Librerías PHP.
Inicialización wearable.	Base de datos.
	Fuente de energía(batería).
	GPRS Shield SIM900.

Fuente: Elaboración propia.

Tabla 7. Tarjeta CRC: Clase ingreso al sistema.

<b>Ingreso al sistema</b>	
Llenar campos de verificación.	Librerías PHP.
Verificación del usuario.	Base de datos.

Fuente: Elaboración propia.

Tabla 8. Tarjeta CRC: Clase captura del ritmo cardiaco.

<b>Captura del ritmo cardiaco</b>	
Captura de los valores del sensor cardiaco.	Sensor cardiaco. IDE arduino.
Ingreso al algoritmo desarrollado en el IDE de arduino.	Arduino nano. Fuente de energía(batería).

*Fuente: Elaboración propia.*

Tabla 9. Tarjeta CRC: Procesamiento de los valores.

<b>Procesamiento de los valores</b>	
Captura valores del ritmo cardiaco en el sensor cardiaco.	IDE arduino Sensor cardiaco.
Determinar los valores del BPM	Arduino nano.

*Fuente: Elaboración propia.*

Tabla 10. Tarjeta CRC: Clase Envió de datos al servidor.

<b>Envío de datos al servidor</b>	
Adquisición del BPM.	Sensor cardiaco
Conectividad a la red móvil.	Librerías PHP.
Conexión con el servidor web.	Base de datos.
Envío al servidor web.	Fuente de energía(batería). IDE arduino.

*Fuente: Elaboración propia.*

Tabla 11. Tarjeta CRC: Clase alerta.

<b>Alerta</b>	
Ingreso del BPM al algoritmo de comprobación. Llamada al cardiólogo cuando exista una variación del BPM.	Wearable. IDE arduino.

*Fuente: Elaboración propia.*

Tabla 12. Tarjeta CRC: Clase monitoreo.

<b>Monitoreo</b>	
Visualización de datos en tiempo real en el sistema web Redacción de observación para el evento del cardiólogo cuando surjan anomalías cardiacas.	Librerías PHP. Base de datos. Wearable. ThingSpeak.

*Fuente: Elaboración propia.*

### 4.1.3. Desarrollo.

Como se mencionó en el proyecto se procederá a presentar la codificación del sistema bajo el patrón de diseño Modelo Vista Controlador(MVC), el cual nos permitía comprender fácilmente y mantener ordenado la codificación del sistema, cabe mencionar que la programación para la placa arduino es netamente estructural, ya que no se puede aplicar la programación orientada a objetos.

Tal y como se mencionó en la primera fase que es la Planificación existen tres iteraciones a continuación se mostrará el desarrollo la primera iteración correspondiente a las 3 historiarías de usuario las cuales son:

- Control de ingreso al sistema.
- Funcionalidad del INTCAP.
- Envío del ritmo cardiaco.

Para la implementación de toda la interfaz gráfica en general se utilizó el editor de código fuente Visual Studio Code, perteneciente a la compañía Microsoft en la cual nos permitió desarrollar con facilidad los distintos componentes del sistema web, primeramente, se estipuló el patrón arquitectónico de desarrollo de software Modelo Vista Controlador(MVC) a partir de ello se empezaron a generar los distintos archivos PHP, HTML, CSS entre otros.

También es necesario mencionar que se utilizaron frameworks y librerías como Bootstrap y JQuery.

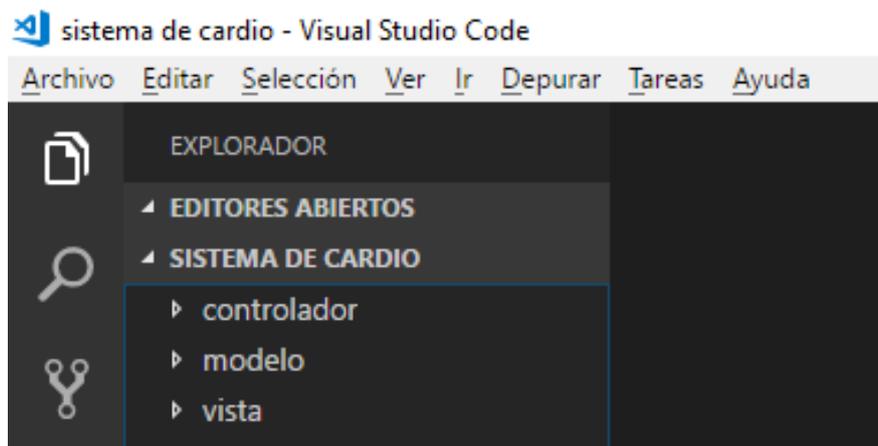


Figura 24. Patrón Modelo Vista Controlador.

Fuente: Elaboración propia.


Figura 25. Control de ingreso al sistema.

Fuente: Elaboración propia.

Ahora con respecto a la funcionalidad del sistema, en la historia de usuario el cliente menciona que en el sistema existan las interfaces necesarias y funcionales que no compliquen el fácil entendimiento del cliente.

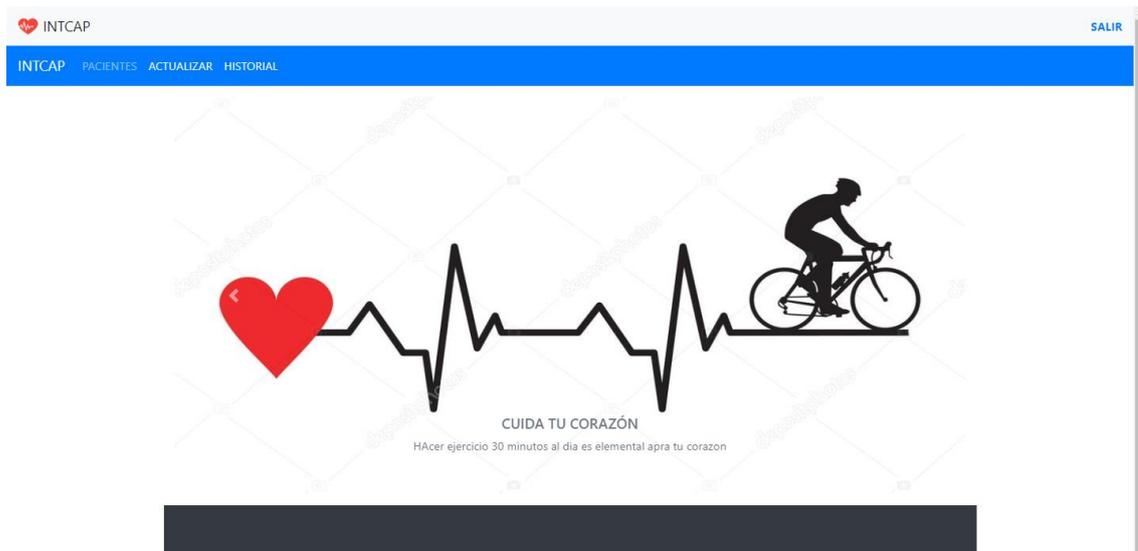


Figura 26. Interfaz principal sistema cardiaco.

Fuente: Elaboración propia.

Ahora en función a la tercera historia de usuario se tiene la codificación en el IDE de arduino con el cual se realizó las pruebas pertinentes con el sensor cardiaco y el módulo Shield SIM900 para su correcto envío a la nube, para lo cual se necesitó el trabajo en tiempo real; dada esta situación se vio con la necesidad de utilizar un servidor web que pudiera rendir para esta historia de usuario.

Para el envío de datos se utilizó el módulo Shield SIM 900 en conjunto con el arduino nano y ThingSpeak que es un servicio de plataforma de análisis de IoT que le permite agregar, visualizar y analizar flujos de datos en vivo en la nube perteneciente a la compañía MathWorks.

```
tesis
//bucle de repeticiones
void loop ()
{
  comandosAT();//Llama a la función comandosAT
  ini_llamada=ini_llamada+1;
  //alerta del sistema :
  if (ini_llamada>10 && (60>BPM || BPM>100)){
    Sim900Serial.println("ATD+51985951660;");
    delay(15000);
    Sim900Serial.println("ATH");
    delay(1000);
  }
}

void comandosAT()
{
  //Sim900Serial.println("AT+CIFSRPT=0");//Establece un indicador '>' al enviar datos
  //serialOutput(); // Metodo de salida (muestra los valores finales).
  //delay(1000);
  mostrarDatosSeriales();
  Sim900Serial.println("AT+CIFSTART=\"TCP\", \"api.thingspeak.com\", \"80\");
  delay(3000);
  mostrarDatosSeriales();
  Sim900Serial.println("AT+CIFSEND");
  delay(2000);
  mostrarDatosSeriales();
  String datos="GET https://api.thingspeak.com/update?api_key=RK4XY4KXKPYTOLH&field1=0"+String(BPM);
  Sim900Serial.println(datos);
  delay(2000);
}
```

Figura 27. Envío de ritmo cardiaco al servidor ThingSpeak.

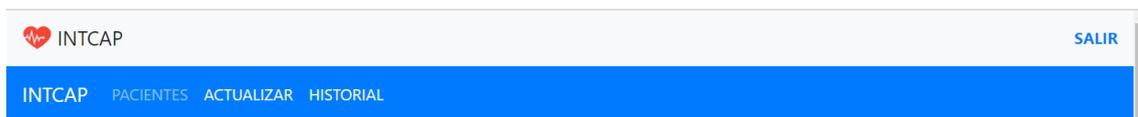
Fuente: Elaboración propia.

Para la segunda iteración se tienen las siguientes historias de usuario.

- Facilidad en el uso
- Monitoreo y visualización del ritmo cardíaco.
- Manejo de datos del cliente.

Para la historia de usuario Facilidad en el uso, se programó la interfaz en función a los requerimientos del cliente y con su constante aceptación ya que según nos manifiesta XP es fundamental que el cliente participe constantemente para asegurar el éxito del proyecto. La interfaz que se desarrolló es intuitiva y de fácil entendimiento.

La interfaz cuenta con 4 menús y un direccionador de link (SALIR), aunque en lo posterior se adicionó uno más.



*Figura 28. Menús del sistema cardíaco.*

*Fuente: Elaboración propia.*

Ahora la presentación del monitoreo del paciente se tuvieron complicaciones en la interconectividad del wearable con el sistema web, pero con las recomendaciones que nos da XP se logró superar sin incidencia mayor alguna. Primeramente, se estableció la comunicación entre el módulo Shield SIM900 a través de sus comandos AT y el servidor web IoT ThingSpeak, seguidamente con el procesamiento del BPM y su envió para su posterior visualización en el sistema web.

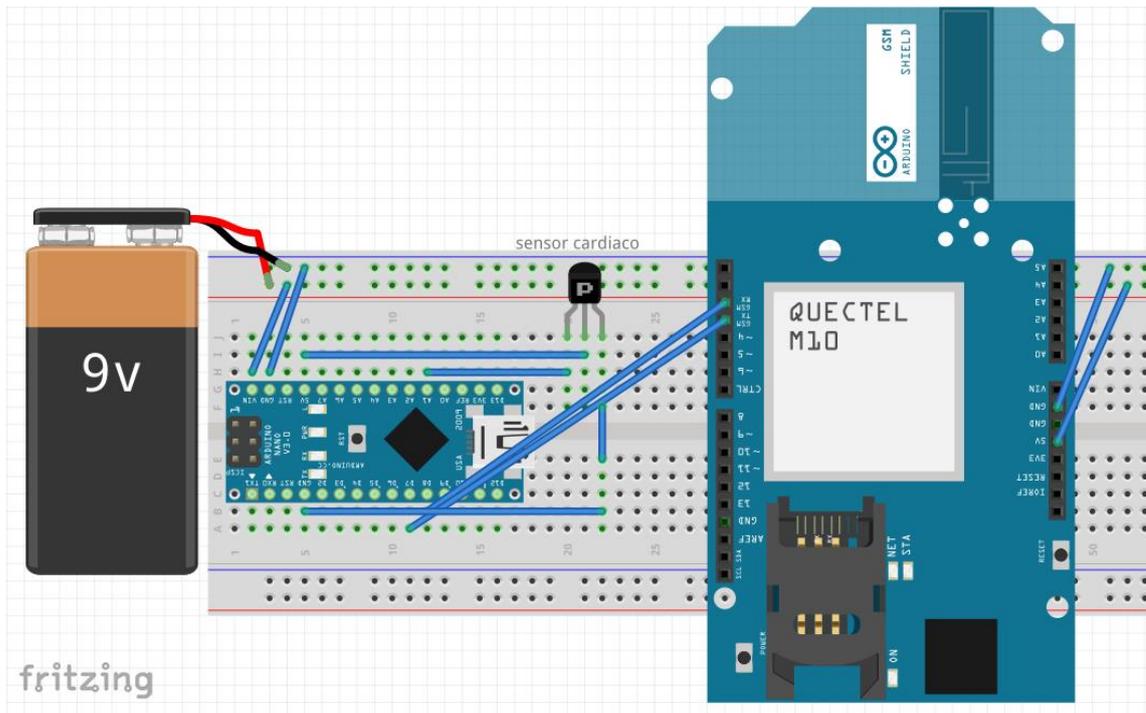


Figura 29. Conexión física de los componentes del wearable elaborado en Fritzing.

Fuente: Elaboración propia.

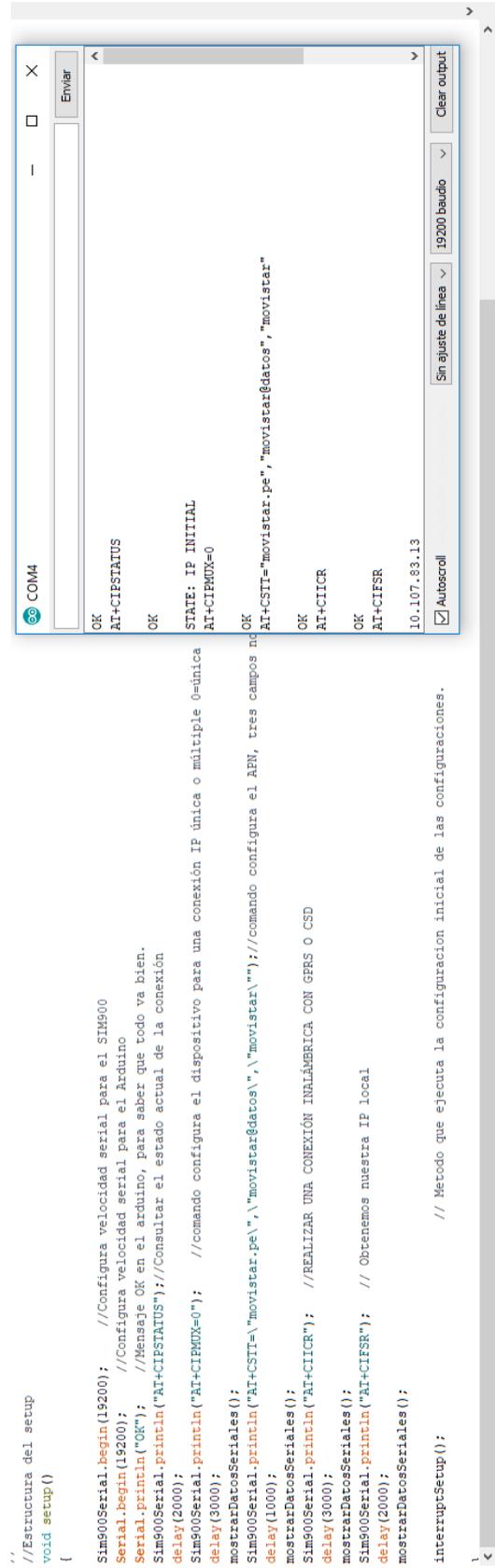


Figura 30. Conexión y envío de BPM del GSM con ThingSpeak.

Fuente: Elaboración propia.

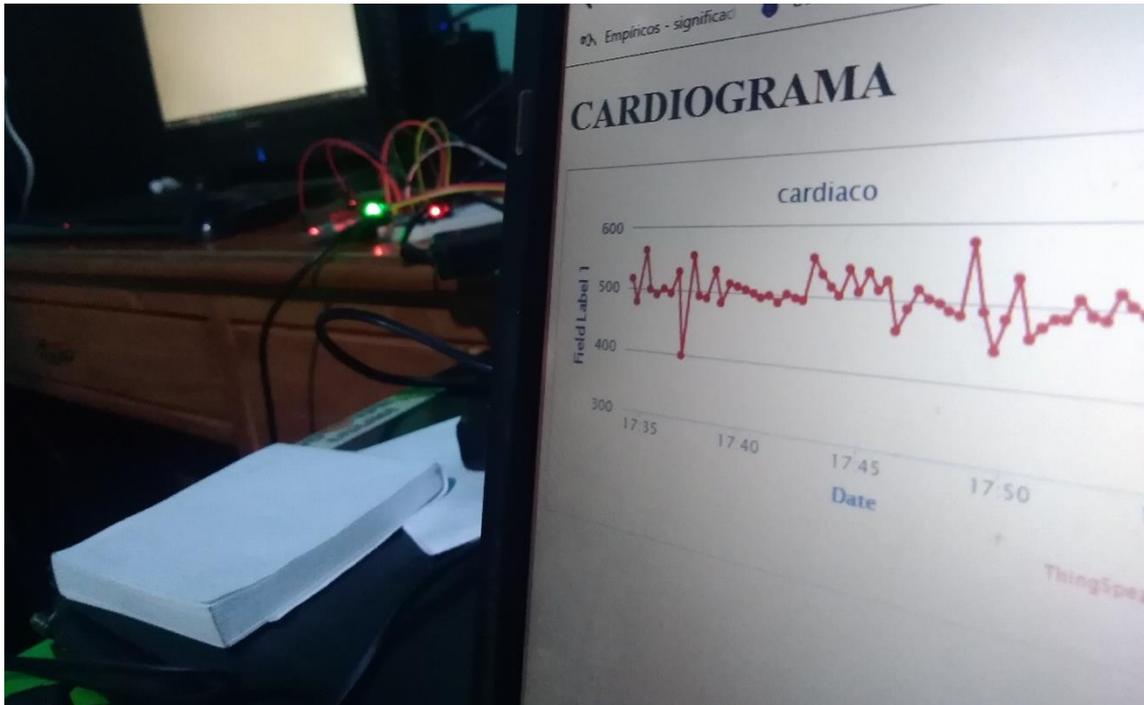


Figura 31. Visualización del ritmo cardiaco en ThingSpeak.

Fuente: Elaboración propia.

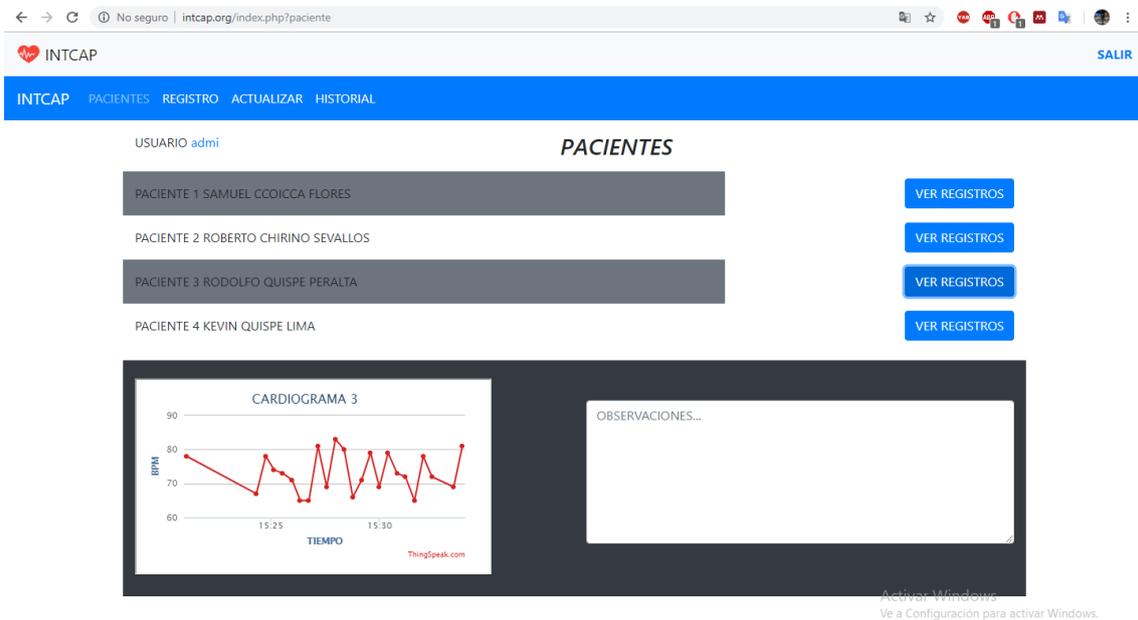


Figura 32. Visualización del ritmo cardiaco en el sistema web de monitoreo.

Fuente: Elaboración propia.

Para el manejo de datos del cliente se desarrolló una base de datos relacional que en interconexión con el sistema cardiaco se puede manejar los datos de este con facilidad; se crearon las tablas que se consideró necesaria para el correcto

funcionamiento y sus respectivas funcionalidades como crear nuevo paciente, actualizar datos y generar historiales de incidencias que se pudieron suscitar.

En esta construcción de la historia se consideró agregar una funcionalidad que es la creación de nuevos clientes que no estuvo estipulada al inicio, así también en esta iteración se modificó el orden de los menús por que se consideró conveniente, para su mejor facilidad de uso.

A continuación, se mostrará la interfaz en el cual existe el manejo de los datos del cliente.

## Actualización

---

Bienvenido!!  
\*Todos los campos son obligatorios

Email	<input type="text"/>
Dirección	<input type="text"/>
Celular	<input type="text"/>
Peso-KG	<input type="text"/>
Talla-CM	<input type="text"/>
Alergias?	<input type="text"/>
Observación	<input type="text"/>

**ACTUALIZAR**

*Figura 33. Actualización de datos del paciente.*

*Fuente: Elaboración propia.*

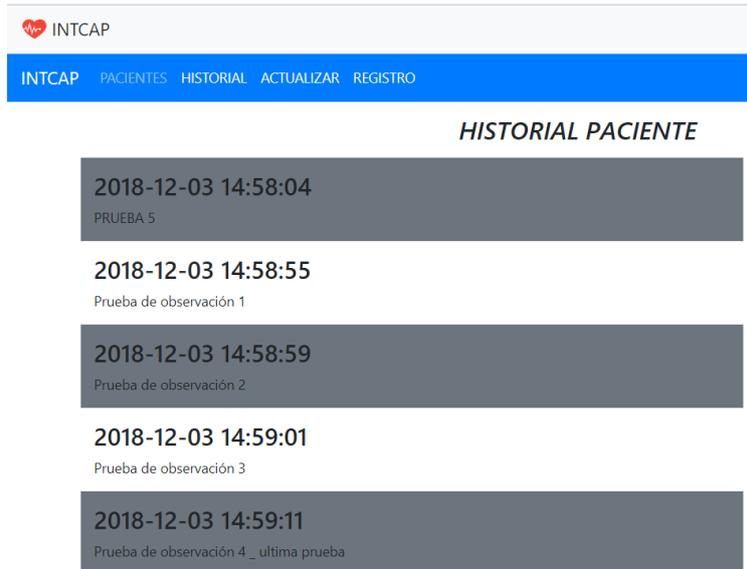


Figura 34. Historial de observaciones del paciente.

Fuente: Elaboración propia.

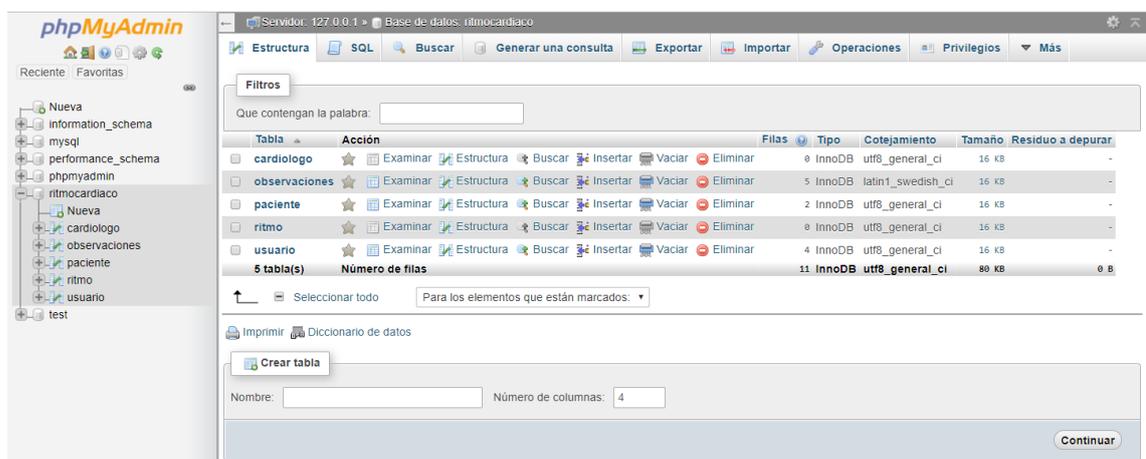
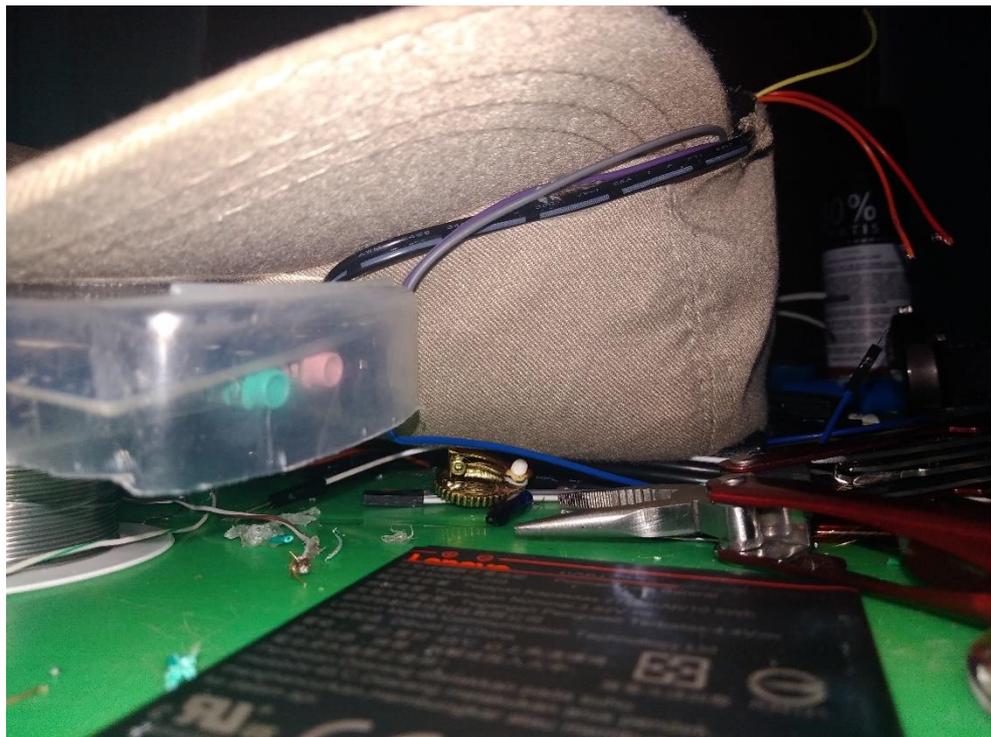


Figura 35. Administrador de la base de datos.

Fuente: Elaboración propia.

Para esta última iteración fueron historias de usuario más basados a la usabilidad del wearable como el sistema web así que la integración de los distintos componentes de hardware. Se vio de forma más conveniente, así como está estipulado en el proyecto la creación de un gorro que contuviera los componentes integrados para la el monitoreo y alerta del ritmo cardiaco, el procedimiento para la integración fue la siguiente.

En la parte superior del gorro se colocó el arduino nano dado su tamaño reducido y adaptabilidad dentro del gorro, seguidamente se cablearon hacia las distintas salidas en las cuales se encontraron conectados el sensor cardiaco, el módulo Shield SIM900 y por ultimo un interruptor para el encendido del wearable conjuntamente con una batería de 9 voltios de salida. Posteriormente después de integrar y adherir estos componentes al gorro se procedió a recubrir internamente el gorro y así no sea visible la estructura cableada del wearable.



*Figura 36. Integración de los componentes del wearable en el gorro.*

*Fuente: Elaboración propia.*



*Figura 37. Recubrimiento interno de la gorra.*

*Fuente: Elaboración propia.*

Para la segunda iteración se tienen las siguientes historias de usuario.

- Alerta de problemas cardiacos.
- Visualización en distintos dispositivos.

La codificación para la alerta cardiaca es simple de comprender, existe una comparativa del BPM en rangos de los cuales no puede exceder ni decrementar en el caso de que pase este suceso, inmediatamente se procederá a llamar al cardiólogo desde el módulo Shield SIM900. El código implementado en el arduino nano es el siguiente.

```

void loop()
{
comandosAT();//Llama a la función comandosAT
ini_llamada=ini_llamada+1;
//alerta del sistema :)
if(ini_llamada>=10 && (60>BPM || BPM>100)){
  sim900Serial.println("ATD+51985951660;");
  delay(15000);
  sim900Serial.println("ATH");
  delay(1000);
}
}

```

Figura 39. Código fuente de la alerta.

Fuente: Elaboración propia.

Con respecto al sistema web se dieron uso de las herramientas existentes en el framework Bootstrap para su facilidad de uso, ubicación de los menús y las distintas tablas y lo necesario que se aplica hoy en día que es la responsividad, esto significa que el sistema web es visible en sus dimensiones ya sea en un dispositivo móvil como Smartphone o Tablet y computadoras de escritorio o también en las portátiles.

#### 4.1.4. Pruebas.



Figura 38. Sistema web responsivo.

Fuente: Elaboración propia.

XP nos manifiesta que las pruebas deben ser desarrolladas en función a las historias de usuario. Las pruebas que se realizaron son unas de las que recomienda XP, pero sin embargo esto no implica que no se puedan realizar otras, a lo largo del transcurso del desarrollo del proyecto se hicieron pequeñas pruebas de funcionalidad de algoritmos ya sea los componentes del wearable así también como en el sistema web. En esta parte se mostrará si realmente lo deseado en las historias de usuario es lo que se implementó y si el sistema y el wearable cumple con lo estipulado desde un inicio por el cliente.

Las iteraciones fueron fundamentales, ya que en las entregas que se estipularon y realizaron fueron aprobadas por el cliente así también ayudaron a solucionar pequeños errores que no tuvieron relevancia algún para el correcto funcionamiento del wearable y el sistema web.

A continuación, se procederá a presentar las pruebas de aceptación que se desarrolló en conjunto con el grupo de trabajo que incluye al cliente.

Tabla 13. Prueba de aceptación: Acceso al sistema con los datos correctos.

Prueba de Aceptación	
<b>Número:1</b>	<b>Historia de Usuario (1):</b> Control de ingreso al sistema.
<b>Nombre:</b> Acceso al sistema con los datos correctos.	
<b>Descripción:</b> Se ingresa al sistema web proporcionando su correo electrónico y la contraseña con la cual se registró, se validan estos datos y se procede a permitir el ingreso al sistema web.	
<b>Condiciones de Ejecución:</b> Conexión a Internet, servidor web, base de datos activa y usuario registrado.	
<b>Entrada / Pasos de ejecución:</b> El usuario ingresa su correo electrónico y su respectiva contraseña creada en su debido momento, a continuación la interfaz de autenticación procesa el pedido y redirecciona a la interfaz principal del sistema web.	
<b>Resultado Esperado:</b> Control de ingreso satisfactorio.	
<b>Evaluación de la Prueba:</b> Prueba satisfactoria.	

*Fuente: Elaboración propia.*

Tabla 14. Prueba de aceptación: Correcto funcionamiento del wearable.

Prueba de Aceptación	
<b>Número:</b> 2	<b>Historia de Usuario (2):</b> Funcionalidad del INTCAP.
<b>Nombre:</b> Correcto funcionamiento del wearable.	
<b>Descripción:</b> Prenda fácil de utilizar sin mucha incomodidad para el paciente, y que no influya en sus actividades cotidianas.	
<b>Condiciones de Ejecución:</b> Batería cargada del wearable, Conexión a Internet, Captura del ritmo cardiaco.	
<b>Entrada / Pasos de ejecución:</b> El paciente procederá a colocarse el gorro con los distintos dispositivos para ver la comodidad de uso.	
<b>Resultado Esperado:</b> Funcionalidad del INTCAP satisfactorio.	
<b>Evaluación de la Prueba:</b> Prueba satisfactoria.	

Fuente: Elaboración propia.

Tabla 15. Prueba de aceptación: Correcto envío del ritmo cardiaco.

Prueba de Aceptación	
<b>Número:</b> 3	<b>Historia de Usuario (3):</b> Envío del ritmo cardiaco.
<b>Nombre:</b> Correcto envío del ritmo cardiaco.	
<b>Descripción:</b> Captura de los valores del sensor cardiaco su correcto procesamiento y envío al sistema web.	
<b>Condiciones de Ejecución:</b> Wearable encendido, Conexión a Internet, Captura del ritmo cardiaco, conexión del SIM900.	
<b>Entrada / Pasos de ejecución:</b> el paciente procede a ponerse la gorra seguidamente prender el wearable con un pulsador de encendido y apagado, prender el módulo Shield SIM900 y por ultimo colocarse el sensor cardiaco en el pabellón del oído derecho.	

<b>Resultado Esperado:</b> Envío del ritmo cardiaco correcto.
<b>Evaluación de la Prueba:</b> Prueba satisfactoria.

*Fuente: Elaboración propia.*

*Tabla 16. Prueba de aceptación: corroboración en la facilidad de uso.*

<b>Prueba de Aceptación</b>	
<b>Número:</b> 4	<b>Historia de Usuario (4):</b> Facilidad en el uso.
<b>Nombre:</b> Corroboración en la facilidad del uso.	
<b>Descripción:</b> visualización y comprobación la funcionalidad y manejo fácil del sistema web así como del wearable (INTCAP).	
<b>Condiciones de Ejecución:</b> Wearable encendido, Conexión a Internet, Captura del ritmo cardiaco, conexión del SIM900, servidor web, base de datos activa y usuario registrado.	
<b>Entrada / Pasos de ejecución:</b> El usuario procederá a dar uso del sistema web generado hasta la iteración presente, primero se autentico posteriormente navegó y revisó cada uno de los menús así como también el uso del wearable.	
<b>Resultado Esperado:</b> Satisfacción del usuario al manejo del sistema web y wearable.	
<b>Evaluación de la Prueba:</b> Prueba satisfactoria.	

*Fuente: Elaboración propia.*

*Tabla 17. Prueba de aceptación: Correcto monitoreo y visualización del ritmo cardiaco.*

<b>Prueba de Aceptación</b>	
<b>Número:</b> 5	<b>Historia de Usuario (5):</b> Monitoreo y visualización del ritmo cardiaco.
<b>Nombre:</b> Correcto monitoreo y visualización del ritmo cardiaco.	

**Descripción:** Se logra ingresar al sistema web con la autenticación posteriormente al monitoreo y ahí la existencia de la visualización del ritmo cardíaco.

**Condiciones de Ejecución:** Wearable encendido, Conexión a Internet, Captura del ritmo cardíaco, conexión del SIM900, algoritmo indexado en el arduino uno, servidor web, base de datos activa y usuario registrado.

**Entrada / Pasos de ejecución:** El usuario procederá a dar uso del sistema web, primero se autentica posteriormente ingreso al menú de monitoreo, luego selecciono el paciente que desea monitorear luego pudo visualizar como resultado el ritmo cardíaco del paciente.

**Resultado Esperado:** Monitoreo del ritmo correcto con datos en tiempo real y su respectiva visualización.

**Evaluación de la Prueba:** Prueba satisfactoria.

*Fuente: Elaboración propia.*

Tabla 18. Prueba de aceptación: Manejo de los datos del cliente.

Prueba de Aceptación	
<b>Número:6</b>	<b>Historia de Usuario (6):</b> Manejo de datos del cliente.
<b>Nombre:</b> Manejo de los datos del cliente.	
<b>Descripción:</b> El cliente tiene la opción de poder actualizar y registrar nuevos usuarios así como manejar sus respectivas observaciones cuando realice el monitoreo.	
<b>Condiciones de Ejecución:</b> Conexión a Internet, servidor web, base de datos activa y usuario registrado, paciente registrado.	
<b>Entrada / Pasos de ejecución:</b> El usuario se autentica y posteriormente recurre al menú de actualizar datos, hace las modificaciones que el considere necesarias y su posterior actualización a través de un botón existente en la interfaz, también ingresando a la opción de registro se pudo ingresar nuevos pacientes para poder realizar su posterior alerta y monitoreo.	
<b>Resultado Esperado:</b> Correcta actualización de datos y registro de nuevos pacientes.	
<b>Evaluación de la Prueba:</b> Prueba satisfactoria.	

*Fuente: Elaboración propia.*

Tabla 19. Prueba de aceptación: Alerta de problemas cardiacos.

Prueba de Aceptación	
<b>Número:</b> 7	<b>Historia de Usuario (7):</b> Alerta de problemas cardiacos.
<b>Nombre:</b> Alerta de problemas cardiacos.	
<b>Descripción:</b> Cuando exista anomalías en el correcto rango que se considera el ritmo cardiaco el wearable realiza llamada al cardiólogo para que realice el monitoreo en tiempo real.	
<b>Condiciones de Ejecución:</b> Wearable encendido, Conexión a Internet, Captura del ritmo cardiaco, conexión del SIM900, algoritmo indexado en el arduino uno, servidor web, base de datos activa, usuario registrado y wearable en uso por el paciente.	
<b>Entrada / Pasos de ejecución:</b> inicialización del wearable, se pone el paciente con sus distintos componentes, se realiza la conexión y su espera respectiva para su estabilidad del sensor, procesamiento de los valores del sensor cardiaco, comparación en el algoritmo, llamada en el caso que exista anomalías en el correcto funcionamiento.	
<b>Resultado Esperado:</b> Alerta de problemas cardiacos de forma correcta.	
<b>Evaluación de la Prueba:</b> Prueba satisfactoria.	

*Fuente: Elaboración propia.*

Tabla 20. Prueba de aceptación: Corroboración de la visualización en distintos dispositivos.

Prueba de Aceptación	
<b>Número:</b> 8	<b>Historia de Usuario (8):</b> Visualización en distintos dispositivos.
<b>Nombre:</b> Corroboración de la visualización en distintos dispositivos.	
<b>Descripción:</b> El sistema debe de ser responsivo para la visualización en dispositivos de escritorio y dispositivos móviles.	
<b>Condiciones de Ejecución:</b> Wearable encendido, Conexión a Internet, Captura del ritmo cardiaco, conexión del SIM900, algoritmo indexado en el arduino uno, servidor web, base de datos activa, usuario registrado y wearable en uso por el paciente.	

**Entrada / Pasos de ejecución:** inicialización del wearable, se pone el paciente con sus distintos componentes, se realiza la conexión y su espera respectiva para su estabilidad del sensor, procesamiento de los valores del sensor cardiaco y envío para su posterior visualización en los distintos dispositivos que tengan conexión a internet.

**Resultado Esperado:** Visualización en los distintos dispositivos de forma correcta.

**Evaluación de la Prueba:** Prueba satisfactoria.

*Fuente: Elaboración propia.*

## CONCLUSIONES Y RECOMENDACIONES

### 5.1. Conclusiones

Hoy en día el uso de IoT a través de los wearables es cada vez más latente y la preocupación de las personas por estar protegidas desde todo punto de vista, y aún más en salud es un punto clave para el desarrollo tecnología en estos aspectos.

Se realizó una búsqueda ampliada de información concerniente a la presente investigación las cuales coadyuvaron positivamente para el desarrollo integral del sistema cardiaco y la creación del wearable.

La investigación fue muy enriquecedora en la adquisición de nuevos conocimientos así fortaleciendo el nivel cognitivo del investigador en aspectos de electrónica, programación, metodologías ágiles de programación, arquitectura de programación, cardiología entre otros aspectos que influyeron para la ejecución del presente proyecto.

Después de desarrollar el presente proyecto se llegó a las siguientes conclusiones.

- El desarrollo del wearable elaborado con arduino, Shield SIM900 y el sensor cardiaco conjuntamente con el sistema web nos permiten monitorear el ritmo cardiaco de las personas que tienen problemas cardiacos y en el caso de que exista anomalías alertar al cardiólogo para que pueda tomar las acciones correspondientes.
- Obtención de la correcta planificación del wearable y del sistema web de monitoreo en función metodología que se utilizó la cual fue Programación Extrema(XP); las historias de usuario jugaron un rol muy importante ya que a partir de ello se fue concibiendo la idea en función a los problemas cardiacos y los requerimientos del cliente.
- La metodología XP y sus distintas herramientas son una buena alternativa para potenciar el desarrollo de un sistema informático y en conjunto con software de diseño orientado a web ayudaron a desarrollar de forma correcta el diseño del sistema web de monitoreo, así también considerando la flexibilidad y simplicidad con la que recomienda trabajar XP se logró bosquejar el wearable.
- Se desarrolló correctamente sin muchas incidencias la codificación respetando la arquitectura de desarrollo Modelo Vista Controlador que a su vez fue de gran ayuda para el trabajo ordenado y organizado del sistema web de monitoreo; en función al wearable con leves dificultades se logró la implementación del código fuente para el arduino nano y su posterior integración con sus componentes complementarios y construirlo en el gorro.
- Las pruebas en conjunto con el cliente fueron fundamentales ya que en cada iteración nos permitieron verificar y realizar la correcta implementación del sistema web reduciendo riesgos y amenazas que perjudicarían la correcta implementación del wearable y sistema web de monitoreo.

## 5.2. Recomendaciones

En el transcurso del desarrollo de la presente investigación se lograron encontrar puntos clave que serán de utilidad a los lectores.

- Antes de iniciar una investigación tecnológica se debe tener en principio el conocimiento necesario sobre las tecnologías que se vayan a utilizar, y contar con el asesoramiento de profesionales con relación al tema a desarrollar, ya que una investigación por más orientada a ingeniería que sea, siempre abarca ámbitos que están fuera de nuestro conocimiento recibido en pregrado.
- Trabajar en las últimas versiones de los distintos softwares de apoyo que se utilice, ya que la compatibilidad con las herramientas que coadyuvan al desarrollo del proyecto puede generar incompatibilidad.
- Al utilizar la metodología XP deben tener una capacitación previa para a razón de que exista comprensión en el desarrollo de la investigación.
- Es fundamental que exista comunicación con el cliente y el equipo de trabajo (si este existiera) para que no exista confusión en el desarrollo del sistema que se vaya a ejecutar.
- Tener especial cuidado con los artefactos que se adquieran para el desarrollo de una investigación y aún más si es electrónico, cerciorarse de su funcionalidad y el ámbito en el que trabaja.
- Esta investigación es un paso más para potenciar el IoT, las personas que deseen continuar la presente investigación o tomarlo como referencia deben tener en cuenta que aún se puede mejorar y generar mejoras para llegar al óptimo rendimiento de este tipo de tecnologías.
- La presente investigación es abierta a obtener mejoras, como el incremento de más componentes para poder tener mayor precisión en la adquisición de signos vitales.
- Las posteriores investigaciones que usen como base conceptual la presente investigación.

## REFERENCIAS BIBLIOGRÁFICAS

- @mduo, & @fat. (n.d.). Bootstrap · The most popular HTML, CSS, and JS library in the world. Recuperado 4 de Diciembre, 2018, de <https://getbootstrap.com/>
- Alejandro Escalante Espinosa. (2013). Estructura y Elementos | Programacion Extrema. Recuperado 4 de Julio, 2018, de <http://isprogramacionextrema.blogspot.com/2013/09/estructura-y-elementos.html>
- ARDUINO. (2018). Arduino GSM Shield. Recuperado 3 de junio, 2018, de <https://store.arduino.cc/usa/arduino-gsm-shield>
- Barea, A. (2016). Diseño, desarrollo y test de un prototipo de pulsera para adquirir el electrocardiograma y la onda de pulso, 96.
- Becerra, Furnieles, G. (2016). Tesis de grado, 1–212.
- Benjamin, E., Blaha, M., Chiuve, S., Cushman, M., Das, S., Deo, R., ... Muntner, P. (2017). Resumen de estadísticas de 2017 Enfermedad del corazón y ataque cerebral. *Circulation. Comité de Estadísticas de Ataque Cerebral de La American Heart Association.*, 2–7. <https://doi.org/10.1161/CIR.0000000000000485>
- Cad, H. (n.d.). Capítulo 2 Metodologías de diseño de hardware, 14–19.
- Das, S., Pal, S., & Mitra, M. (2017). Arduino-based noise robust online heart-rate detection. *Journal of Medical Engineering and Technology*, 41(3), 170–178. <https://doi.org/10.1080/03091902.2016.1271044>
- González, F., Villegas, O., Ramírez, D., Sánchez, V., & Domínguez, H. (2014). Smart Multi-Level Tool for Remote Patient Monitoring Based on a Wireless Sensor Network and Mobile Augmented Reality. *Sensors*, 14(9), 17212–17234. <https://doi.org/10.3390/s140917212>
- Gubbi, J., Buyya, R., Marusic, S., & Palaniswami, M. (2013). Internet of Things ( IoT ): A vision , architectural elements , and future directions. *Future Generation Computer Systems*, 29(7), 1645–1660. <https://doi.org/10.1016/j.future.2013.01.010>
- Guerra, J. L. G., & Perez, A. F. (1847). Implementación de un módulo de consulta y medición de ritmo cardiaco con envío de datos GSM usando Open Hardware, (August 2014).
- INFOBAE. (2016). Por qué las enfermedades cardíacas todavía son la primera causa de muerte en el mundo - Infobae. Retrieved June 3, 2018, from <https://www.infobae.com/salud/2016/09/25/por-que-las-enfermedades-cardiacas-todavia-son-la-primera-causa-de-muerte-en-el-mundo/>
- Ing, P., Gardu, M., Dra, Z. D., & Muhl, S. (2011). Instituto Politécnico Nacional. <https://doi.org/10.1109/CCECE.2014.6901098>
- Learn More - ThingSpeak IoT. (n.d.). Recuperado 4 de Diciembre, 2018, de [https://thingspeak.com/pages/learn\\_more](https://thingspeak.com/pages/learn_more)
- Letelier, P., Canós, M., Sánchez, E., & Penadés, M. (2003). Metodologías Ágiles en el Desarrollo de Software. *Valencia, Valencia, España*, 1–8. Recuperado de [http://www.carlosfau.com.ar/nqi/nqifiles/XP\\_Agil.pdf](http://www.carlosfau.com.ar/nqi/nqifiles/XP_Agil.pdf)
- Medicina General Univesal. (2017). MedicinaGeneralUniversal: Que es un electrocardiograma. Recuperado 4 de junio, 2018, de <http://medicinageneraluniversal.blogspot.com/2017/01/que-es-un-electrocardiograma.html>
- Mendis S, Puska P, Norrving, B. editors, González, R., Alcalá, J., Moran, A. E., Forouzanfar, M. H., Roth, G. A., ... Ministerio de Sanidad, S. S. E. I. (2014). Perfil de enfermedades cardiovasculares. *Revista de La Facultad de Medicina de La UNAM*, 2014(14), 303. <https://doi.org/10.1161/CIRCULATIONAHA.113.004046>
- MINSA. (2016). .: MINSA.: - Ministerio de Salud del Perú. Recuperado 5 de Junio, 2018, de <http://www.minsa.gob.pe/?op=51&nota=20847>
- My EKG. (n.d.). Intervalos y Segmentos del Electrocardiograma. Recuperado 4 de Junio, 2018, de <http://www.my-ekg.com/generalidades-ekg/intervalos-segmentos-ekg.html>
- Organización Mundial de la Salud(OMS). (2017). Enfermedades cardiovasculares. Recuperado 3 de Junio, 2018, de <http://www.who.int/es/news-room/fact->

- sheets/detail/cardiovascular-diseases-(cvds)
- Ortiz Lima, F. T. (2016). Diseño de un sistema inalámbrico para monitoreo de pacientes ambulatorios, utilizando sensores de presión arterial y ritmo cardíaco e implementación de un prototipo de prueba.
- Park, S., & Jayaraman, S. (2003). Enhancing the Quality of Life Through Wearable Technology. *IEEE Engineering in Medicine and Biology Magazine*, 22(3), 41–48. <https://doi.org/10.1109/MEMB.2003.1213625>
- Pawar, P. A. (2014). Heart rate monitoring system using IR base sensor & Arduino Uno. *Proceedings of the 2014 Conference on IT in Business, Industry and Government: An International Conference by CSI on Big Data, CSIBIG 2014*, 1–3. <https://doi.org/10.1109/CSIBIG.2014.7057005>
- Peets, A. (2013). GUÍA PARA EL DESARROLLO ÁGIL DE APLICACIONES EN ROBÓTICA. (Spanish). *METHODS AND TOOLS USED IN FAST DEVELOPMENT OF ROBOTICS PROJECTS. (English)*, 8(16), 136–147. Recuperado de <http://search.ebscohost.com/login.aspx?direct=true&db=fua&AN=93913464&lang=es&site=ehost-live>
- Perú21. (2017). Enfermedades al corazón: Informe revela todo lo que gasta el Perú y cómo se pueden evitar | Lima | Peru21. Recuperado 3 de Junio, 2018, de <https://peru21.pe/lima/enfermedades-corazon-informe-revela-gasta-peru-evitar-68227>
- Sierra, M. (2008). Qué es un servidor y cuáles son los principales tipos de servidores (proxy,dns, web,ftp,pop3 y smtp, dhcp...). *Aprenderaprogramar.Com*, 2. Recuperado de <https://www.aprenderaprogramar.com/attachments/article/542/DV00408A> Que es un servidor principales tipos proxy smtp ftp web dns.pdf%0Ahttp://aprenderaprogramar.com/index.php?option=com\_content&view=article&id=542:que-es-un-servidor-y-cuales-son-los-princi
- Silva, D. R., Douglas, F., Nakamura, E. H., Oliveira, M. T., & Júnior, J. A. J. (2014). Desenvolvimento de um Oxímetro de baixo custo utilizando Arduino e Smartphone Development of a low cost Oximeter using Arduino and Smartphone, 1–4.
- Technology, C., Nusantara, U. M., & Serpong, G. (2016). Department of Computer Engineering, Faculty of Information and Communication Technology, Universitas Multimedia Nusantara, Jl. Scientia Boulevard, Gading Serpong, Tangerang 15810, Indonesia (Received: August 2015 / Revised: January 2016 / Accepted: Januar, 78–87.
- Técnica administrativa*. (2002). Ciencia y Técnica Administrativa. Recuperado de <http://www.cyta.com.ar/ta0502/v5n2a1.htm>
- Uribe, W., Duque, M., & Medina, E. (2008). Electrocardiografía y arritmias. *Vasa*, 144. Retrieved from <http://medcontent.metapress.com/index/A65RM03P4874243N.pdf>
- Velázquez Elizondo, P. E., & Villagrán Valencia, L. J. (2017). Monitor de ritmo cardíaco para dispositivos android mediante un enlace bluetooth. *Instituto Politécnico Nacional*, 119. Recuperado de <http://tesis.ipn.mx/handle/123456789/20455?show=full>
- VITAL. (2014). Enfermedades cardiovasculares causan el 15.5% de muertes en el Perú. Recuperado 3 de Junio, 2018, de <http://vital.rpp.pe/salud/enfermedades-cardiovasculares-causan-el-155-de-muertes-en-el-peru-noticia-692924>

## ANEXOS

### 7.1. Historias de usuario

Historia de Usuario	
Numero: 1	Nombre Historia Usuario: Control de ingreso al sistema
Usuario: Administrador (cardiologo)	
Modificación de Historia N°:	Iteración Asignada: 1
Prioridad en Negocio: (Alta/Media/Baja) Media	Puntos Estimados: 6
Riesgo en Desarrollo: (Alta/Media/Baja) Baja	Puntos Reales: 5
Descripción: Permitir el ingreso a los usuarios que se encuentren debidamente registrados.	
Observaciones:	

Historia de Usuario	
Numero: 2	Nombre Historia Usuario: Funcionalidad del INTICAP
Usuario: Paciente	
Modificación de Historia N°: -	Iteración Asignada: 1
Prioridad en Negocio: (Alta/Media/Baja) Alta	Puntos Estimados: 9
Riesgo en Desarrollo: (Alta/Media/Baja) Media	Puntos Reales: 9
Descripción: Artefacto que tenga facilidad en su uso, debe de ser Prenda de uso cotidiano y que no incomode en el desarrollo de las actividades	
Observaciones:	

Historia de Usuario	
Numero: 3	Nombre Historia Usuario: Envío del ritmo cardíaco
Usuario: Administrador (cardiólogo)	
Modificación de Historia N°: -	Iteración Asignada: 1
Prioridad en Negocio: (Alta/Media/Baja) Alta	Puntos Estimados: 9
Riesgo en Desarrollo: (Alta/Media/Baja) Media	Puntos Reales: 8
Descripción: El artefacto desarrollado (wearable) debe de capturar los valores reales del paciente para su procesamiento y posterior envío	
Observaciones:	

Historia de Usuario	
Numero: 4	Nombre Historia Usuario: Facilidad en el uso
Usuario: Administrador (cardiólogo)	
Modificación de Historia N°: -	Iteración Asignada: 2
Prioridad en Negocio: (Alta/Media/Baja) Media	Puntos Estimados: 7
Riesgo en Desarrollo: (Alta/Media/Baja) Baja	Puntos Reales: 6
Descripción: El sistema web debe ser fácil de usar así como también fácil para monitorear al paciente.	
Observaciones:	

Historia de Usuario	
Numero: 5	Nombre Historia Usuario: Monitoreo y visualización del ritmo cardiaco
Usuario: Administrador (cardiologo)	
Modificación de Historia N°:	Iteración Asignada: 2
Prioridad en Negocio: (Alta/Media/Baja) Alta	Puntos Estimados: 9
Riesgo en Desarrollo: (Alta/Media/Baja) Media	Puntos Reales: 9
Descripción: El sistema debe permitir monitorear en tiempo real al paciente, en una interfaz mostrar los resultados y los valores del BPM obtenido.	
Observaciones:	

Historia de Usuario	
Numero: 6	Nombre Historia Usuario: Manejo de datos del cliente
Usuario: Administrador (cardiologo)	
Modificación de Historia N°: -	Iteración Asignada: 2
Prioridad en Negocio: (Alta/Media/Baja) Media	Puntos Estimados: 6
Riesgo en Desarrollo: (Alta/Media/Baja) Baja	Puntos Reales: 5
Descripción: Actualización de los datos del paciente, por que existe variación en el peso y la talla y algunas alergias que puede desarrollar	
Observaciones:	

Historia de Usuario	
Numero: 7	Nombre Historia Usuario: Alerta de problemas cardiacos
Usuario: Administrador (Cardiologo), Paciente	
Modificación de Historia N°:	Iteración Asignada: 3
Prioridad en Negocio: (Alta/Media/Baja) Alta	Puntos Estimados: 10
Riesgo en Desarrollo: (Alta/Media/Baja) Media	Puntos Reales: 9
Descripción: <p>cuando exista variación en el ritmo normal, se debe de proceder a realizar llamada al cardiologo para que tome medidas dependiendo a la monitorización en tiempo real.</p>	
Observaciones:	

Historia de Usuario	
Numero: 8	Nombre Historia Usuario: Visualización en distintos dispositivos
Usuario: Administrador (cardiologo)	
Modificación de Historia N°: -	Iteración Asignada: 3
Prioridad en Negocio: (Alta/Media/Baja) Baja	Puntos Estimados: 5
Riesgo en Desarrollo: (Alta/Media/Baja) Baja	Puntos Reales: 4
Descripción: <p>el sistema web debe ser responsivo, para su visualización en cualquier lugar donde exista señal y red de internet movil.</p>	
Observaciones:	

## 7.2. Código fuente arduino

```
#include <SoftwareSerial.h>
#include <String.h>
//GPRS/GSM
SoftwareSerial Sim900Serial(7, 8); //Seleccionamos los pines 7 como Rx y 8 como Tx
//-----
//declaración de variables
//sensor cardiaco
volatile int Transcurrido_Array[10]; // Arreglo para registrar el tiempo transcurrido entre cada pulso(IBEI).
volatile unsigned long Respaldo_Conteo = 0; // Contador de respaldo
volatile unsigned long Momento_UltimoPulso = 0; // Tiempo transcurrido hasta el ultimo pulso.
volatile int P = 512; // Punto mas alto de una onda de pulso.
volatile int T = 512; // Punto inicial del pico de onda.
volatile int ValorMedioREF = 530; // Punto medio del pico de onda.
volatile int AnchoOndaValido = 0; // Ancho del pico de onda.
volatile boolean Primer_Pulso = true; // Primera porcion de pulso para el calculo del BPM (arreglo).
volatile boolean Segundo_Pulso = false; // Segunda porcion de pulso para el registro del BPM (arreglo).
int Pin_Entrada_sensor = 1; // Pin A1, donde se recibe la señal del sensor.
// Volatile Variables, es un tipo variable que cuando ya es usada se desecha el valor que contiene y toma uno nuevo.
// Mediante volatile se actualizan los datos tomados
volatile boolean Q_S = false;
volatile int BPM; // Pulsos por Minuto se actualiza constantemente.
volatile int signal_sensor; // Almacena todos los datos enviados por el sensor.
volatile int IBI = 600; // Almacena el tiempo entre cada de pulso (Valor 600 de referencia).
volatile boolean Pulso = false; // Para la detección de pulso valido.
int ini_llamada=0; // Variable que nos ayudara para dar la alerta desde el primer minuto de uso.
//-----
//Estructura del setup
void setup()
{
  Sim900Serial.begin(19200); //Configura velocidad serial para el SIM900
  Serial.begin(19200); //Configura velocidad serial para el Arduino
  Serial.println("OK"); //Mensaje OK en el arduino, para saber que todo va bien.
  Sim900Serial.println("AT+CIPSTATUS");//Consultar el estado actual de la conexión
  delay(2000);
  Sim900Serial.println("AT+CIPMUX=0"); //comando configura el dispositivo para una conexión IP única o múltiple 0=única
  delay(3000);
  mostrarDatosSeriales();
  Sim900Serial.println("AT+CSTT=\"movistar.pe\",\"movistar@datos\",\"movistar\");//comando configura el APN, tres campos nombre APN, Usuario y contraseña->Movistar Perú.
  delay(1000);
  mostrarDatosSeriales();
  Sim900Serial.println("AT+CIICR");// REALIZAR UNA CONEXIÓN INALÁMBRICA CON GPRS O CSD
  delay(3000);
  mostrarDatosSeriales();
  Sim900Serial.println("AT+CIFSR");// Obtenemos nuestra IP local
  delay(2000);
  mostrarDatosSeriales();

  interruptSetup(); // Metodo que ejecuta la configuración inicial de las configuraciones.
}

void interruptSetup(){ //CONFIGURACION
  TIMSK2 = 2; // ACTIVA LAS PAUSA DEL MARCADOR ENTRE EL (timer)
  sei(); // DEFINE GLOBALMENTE LOS PARAMETROS ESTALECIDOS ARRIBA INICIANDOSE
}

//-----
//bucle de repeticiones
void loop()
{
  comandosAT();//Llama a la función comandosAT
  ini_llamada=ini_llamada+1;
  //alerta del sistema :)
  if(ini_llamada==10 && (60-BPM != 100)){
    Sim900Serial.println("ATD=5198551660;");
    delay(15000);
    Sim900Serial.println("ATH");
    delay(1000);
  }
}

void comandosAT()
{
  //Sim900Serial.println("AT+CIPSPRT=0");//Establece un indicador ">" al enviar datos
  //SerialOutput(); // Metodo de salida (muestra los valores finales).
  //delay(1000);
  mostrarDatosSeriales();
  Sim900Serial.println("AT+CIPSTART=\"TCP\",\"api.thingspeak.com\",80");//Indicamos el tipo de conexión, url o dirección IP y puerto al que realizamos la conexión
  delay(3000);
  mostrarDatosSeriales();
  Sim900Serial.println("AT+CIPSEND");//Envia datos a través de TCP o UDP
  delay(2000);
  mostrarDatosSeriales();
  String datos="GET https://api.thingspeak.com/update?api_key=RK4XY4KXKPYTOLH&field=0"+String(BPM); //Envia datos al servidor remoto
  Sim900Serial.println(datos);
  delay(2000);
  mostrarDatosSeriales();
  Sim900Serial.println((char)26);
  delay(5000); //Ahora esperamos una respuesta pero esto va a depender de las condiciones de la red.
}
}
```

```

void serialOutput(){
  Serial.print(BPM); // Valor de pulsos por minuto.
}
//-----
//metodo para hallar y controlar el BPM
ISR(TIM2_COMP_vect){
  cli(); // Detiene el servicio de interrupcion de rutina (ISR).
  signal_sensor= analogRead(Pin_Entrada_sensor); // Lectura del valor arrojado por el sensor.
  Respaldo_Conteo += 2; // Lleva un conteo auxiliar acumulado a 2 por ciclo.
  int N = Respaldo_Conteo - Momento_UltimoPulso; // "N" registra el tiempo despues de el ultimo pulso.
  // encuetra un pico optimo donde empezar los calculos del pulso cardiaco.
  if(signal_sensor< ValorMedioREF && N > (IBI/5)*3){ // Evita los falsos latidos o latidos no validos.
  if (signal_sensor< T){ // SI la lectura del pulso es mayor a 512.
    T = signal_sensor; // Pulso de pico mas bajo.
  }
}
if(signal_sensor> ValorMedioREF && signal_sensor> P){ // Condicion para captar picos superiores (por encima de 530).
P = signal_sensor; // P es el punto mas alto.
}
if (N > 250){ // Condicion cuando han transcurrido mas de 250 ms.
if ( (signal_sensor> ValorMedioREF) && (Pulso == false) && (N > (IBI/5)*3) ){
  Pulso = true; // Se activa para posibles pulsos.
  IBI = Respaldo_Conteo - Momento_UltimoPulso; // Medida entre pico a pico de un pulso en ms.
  Momento_UltimoPulso = Respaldo_Conteo; //Guarda el ultimo pulso en otra variable.
  if(Segundo_Pulso){ // Evalua si se registro un pulso antes (si ya paso por el primero)
    Segundo_Pulso = false; // Pone el valor en falso para evaluar nuevamente la condicion.
    for(int i=0; i<=9; i++){ // Toma todos los 10 valores validos de IBI.
      // los valores de IBI guardados son referencia para el BPM.
      // IBI esta medido en ms y los almacena en una variable.
      // Guarda el valor de los IBI obtenidos.
      Transcurrido_Array[i] = IBI;
    }
  }
  if(Primer_Pulso){ // Evalua el primer pulso y prepara el segundo pulso.
    Primer_Pulso = false; // Cambiamos a false para que evalue nuevamente la condicion.
    Segundo_Pulso = true; // Da aviso que el pulso siguiente es el segundo.
    sei(); // Reanuda la interrupciones.
  }
  word runningTotal = 0; // Guarda la sumatoria de todos los valores del IBI ms.
  for(int i=0; i<=9; i++){ // Ciclo para filtrar los datos de "Transcurrido_Array".
    Transcurrido_Array[i] = Transcurrido_Array[i+1]; // Organizamos el arreglo para dejar solo los valores actuales.
    runningTotal += Transcurrido_Array[i]; // Suma de IBI contenidos en las primeras 9 posiciones del arreglo.
  }
  Transcurrido_Array[9] = IBI; // Ingresa el ultimo valor de IBI.
  runningTotal += Transcurrido_Array[9]; // Agrega el ultimo IBI a runningTotal.
  runningTotal /= 10; // Promedio de los ultimos 10 IBI.
  BPM = 60000/runningTotal; // Cantidad de pulsaciones por minuto.
}
}
if ((signal_sensor < ValorMedioREF) && (Pulso == true)){ // Cuando capture el final de un pulso CARDIACO valido.
  Pulso = false; // Pone el pulso en "false" por que ya esta en el punto final del pulso (S)
  AnchoOndaValido = P - T; // Mide la distancia que existe entre el punto del pulso mas alto y el punto mas bajo de la lectura de pulso
  ValorMedioREF = AnchoOndaValido/2 + T; // una pendiente cuya longitud determina la altura de la onda
P = ValorMedioREF; // Actualiza valores.
T = ValorMedioREF;
}
if (N > 2500){ // Si han pasado 2.5 segundos y no hay un pulso, tome valores iniciales
ValorMedioREF = 530; // ValorMedioREF POR DEFECTO EL MISMO VALOR De INICIO
P = 512; // VARIABLE P Y T a valor 512 como al inicio
T = 512;
Momento_UltimoPulso = Respaldo_Conteo; // Llama el ultimo pulso que exista del contador de muestra
Primer_Pulso = true; // El primer pulso toma el valor inicial por defecto
Segundo_Pulso = false; // El segundo pulso toma el valor inicial por defecto
}
// FIN DEL METODO DE CUANTIFICACION DE DATOS VALIDOS
sei(); }
//-----
//Metodo que nos permite detectar los errores en los distitos comandos AT.
void mostrarDatosSeriales()
{
  while(Sim900Serial.available() !=0)
  Serial.write(Sim900Serial.read());
}
}

```

7.3. Funcionamiento INTCAP



