



UNIVERSIDAD
PRIVADA
DEL NORTE

FACULTAD DE INGENIERÍA

CARRERA DE INGENIERÍA DE SISTEMAS COMPUTACIONALES

“IMPLEMENTACIÓN DE UNA APLICACIÓN MÓVIL
COMPLEMENTADA CON LA TECNOLOGÍA DE
SENSORES MULTIPARAMÉTRICOS CON ARDUINO
PARA MONITOREAR LA CALIDAD DEL AGUA
POTABLE EN EL DISTRITO DE LAREDO, TRUJILLO”

Tesis para optar el título profesional de:

Ingeniero de Sistemas Computacionales

Autor(es):

Br. Ocola Gadea, Raúl Alfonso
Br. Ramos Mendoza, Gorky Walter

Asesor:

Mg. José Gómez Ávila

Trujillo – Perú
2018

APROBACIÓN DE LA TESIS

El (La) asesor(a) y los miembros del jurado evaluador asignados, **APRUEBAN** la tesis desarrollada por el (la) Bachiller **Ocola Gadea Raúl Alfonso; Ramos Mendoza Gorky Walter**, denominada:

**“IMPLEMENTACIÓN DE UNA APLICACIÓN MÓVIL COMPLEMENTADA CON LA
TECNOLOGÍA DE SENSORES MULTIPARAMÉTRICOS CON ARDUINO PARA
MONITOREAR LA CALIDAD DEL AGUA POTABLE EN EL DISTRITO DE LAREDO,
TRUJILLO”**

Mg. José Gómez Ávila
ASESOR

Mg. Víctor Enemesio Dávila Rodríguez
JURADO
PRESIDENTE

Mg. Rolando Javier Berrú Beltrán
JURADO

Mg. Luis Mauricio Gutiérrez Magán
JURADO

DEDICATORIA

A Dios, por habernos permitido llegar hasta este punto y habernos dado salud para lograr nuestros objetivos, además de su infinita bondad y amor.

A nuestros padres por ser el pilar fundamental en todo lo que somos, en toda nuestra educación, tanto académica, como de la vida, por su incondicional apoyo perfectamente mantenido a través del tiempo. Todo este trabajo ha sido posible gracias a ellos.

AGRADECIMIENTO

Gracias a Dios y a nuestras familias por permitirnos cumplir con excelencia en el desarrollo de este proyecto de tesis. No ha sido sencillo el camino hasta ahora, pero gracias a sus aportes, a su amor, a su inmensa bondad y apoyo, lo complicado de lograr esta meta se ha notado menos. Les agradecemos, y hacemos presente nuestro gran afecto hacia ustedes.

De igual forma, agradecemos a nuestro asesor de Tesis, Mg. GÓMEZ AVILA, que gracias a sus consejos y correcciones hoy podemos culminar este trabajo. A los Profesores que nos han visto crecer como personas, y gracias a sus conocimientos hoy podemos sentirnos dichosos y contentos.

ÍNDICE DE CONTENIDOS

APROBACIÓN DE LA TESIS	2
DEDICATORIA.....	3
AGRADECIMIENTO	4
ÍNDICE DE CONTENIDOS	5
ÍNDICE DE TABLAS	7
ÍNDICE DE FIGURAS.....	8
RESUMEN.....	9
ABSTRACT	10
CAPÍTULO 1. INTRODUCCIÓN.....	11
1.1. Realidad problemática	11
1.2. Formulación del problema.....	13
1.3. Justificación.....	13
1.4. Limitaciones	14
1.5. Objetivos	14
1.5.1. <i>Objetivo general</i>	14
1.5.2. <i>Objetivos específicos</i>	14
CAPÍTULO 2. MARCO TEÓRICO	15
2.1. Antecedentes.....	15
2.2. Bases teóricas	17
2.2.1. <i>Parámetros de Calidad del Agua</i>	17
2.2.2. <i>Arduino</i>	22
2.2.3. <i>IONIC (Mobile App Framework)</i>	24
2.2.4. <i>Sensores</i>	25
2.2.5. <i>Metodologías de Desarrollo</i>	31
2.2.5.2. <i>Metodologías de desarrollo tradicionales</i>	35
CAPÍTULO 3. HIPÓTESIS	35
3.2. Formulación de la Hipótesis.....	36
3.3. Operacionalización de variables.....	36
CAPÍTULO 4. DESARROLLO.....	37
4.1. Metodología.....	37
CAPÍTULO 5. METODOLOGÍA.....	75

5.1.	Diseño de investigación	75
	5.1.1. <i>Pre Experimental</i>	75
5.2.	Unidad de estudio.....	75
5.3.	Población.....	75
5.4.	Muestra (muestreo o selección)	75
5.5.	Técnicas, instrumentos y procedimientos de recolección de datos	76
5.6.	Métodos, instrumentos y procedimientos de análisis de datos	76
CAPÍTULO 6.	RESULTADOS	76
CAPÍTULO 7.	DISCUSIÓN.....	92
CONCLUSIONES.....		94
RECOMENDACIONES.....		96
REFERENCIAS		97
ANEXOS		100

ÍNDICE DE TABLAS

Tabla 1 Operacionalización de variables.....	36
Tabla 2 Cuadro comparativo de metodología SCRUM y RUP	37
Tabla 3 Recursos Humanos.....	39
Tabla 4 Presupuesto de Recursos de Hardware y Software	39
Tabla 5 Material de Escritorio.....	40
Tabla 6 Requerimiento no Funcional – Velocidad en el tiempo de respuesta.....	42
Tabla 7 Requerimiento no Funcional - Usabilidad	43
Tabla 8 Requerimiento no Funcional – Restricciones técnicas derivadas del entorno tecnológico	43
Tabla 9 Requerimiento Funcional – CU01_Registrar Parámetros de calidad.....	44
Tabla 10 Requerimiento Funcional – CU02_Consultar Resultados.....	45
Tabla 11 Descripción Arquitectura Física	54
Tabla 12 Descripción de Arquitectura Lógica	55
Tabla 13 Descripción Diagrama de Componentes.....	56
Tabla 14 Descripción Tabla Mont_Agua_Tesis	57
Tabla 15 Descripción Procedimiento almacenado MONT_INSERTAR_RESULTADOS_SP	57
Tabla 16 Descripción Procedimiento Almacenado MONT_LISTAR_RESULTADOS_SP	58
Tabla 17 Técnicas, instrumentos y procedimientos.	76
Tabla 18: Métodos, instrumentos y procedimientos.....	76
Tabla 19 Tabla de Indicadores.....	77
Tabla 20 Escala de Medición Indicadores Cuantitativos	77
Tabla 21 Nivel de completitud funcional sin la aplicación móvil.....	79
Tabla 22 Nivel de completitud Funcional con la Aplicación Móvil.....	80
Tabla 23 Nivel de Corrección Funcional sin aplicación móvil.....	83
Tabla 24 Nivel de Corrección Funcional con aplicación móvil.....	85
Tabla 25 Nivel del comportamiento temporal sin la aplicación móvil.	88
Tabla 26 Nivel de comportamiento temporal con la aplicación móvil.....	89
Tabla 27 Cálculo de Varianza antes de la implementación de aplicación móvil.....	90
Tabla 28 Cálculo de Varianza después de implementación de aplicación móvil.....	90
Tabla 29 Nivel de Impacto – Indicador 1	92
Tabla 30 Nivel de Impacto – Indicador 2	93
Tabla 31 Nivel de Comportamiento Temporal	94

ÍNDICE DE FIGURAS

Figura 1 Límites Máximos permisibles referenciales de los parámetros de calidad del agua potable	17
Figura 2 Escala del pH	20
Figura 3 Comparación de las especificaciones de tarjetas de desarrollo	23
Figura 4 Sensor de Temperatura NTC 3950	26
Figura 5 Sensor de Turbidez SEN0189	27
Figura 6 Diagrama de Conexión de Sensor de Turbidez al Arduino	28
Figura 7 Sensor de Conductividad Eléctrica DFR0300.....	28
Figura 8 Diagrama de Conexión de Sensor de Conductividad Eléctrica DFR0300 al Arduino.....	29
Figura 9 Sensor de pH SEN0161.....	30
Figura 10 Diagrama de Conexión de Sensor de pH SEN0161 al Arduino.....	31
Figura 11 Desarrollo Tradicional vs. Desarrollo Ágil.....	32
Figura 12 Cronograma de actividades	41
Figura 13 Diagrama de Actividades – Registrar parámetros de Calidad	46
Figura 14 Diagrama de Actividades – Consultar Resultados	47
Figura 15 Diagrama de Secuencia – Registrar Parámetros de Calidad	48
Figura 16 Diagrama de Secuencia – Consultar Resultados.....	49
Figura 17 Interfaz de Pantalla inicial de Aplicación Móvil	50
Figura 18 Interfaz de resultados de monitoreo e Información de la aplicación móvil	51
Figura 19 Interfaz de Lista de Resultados de Monitoreo.....	52
Figura 20 Interfaz de Resultado del Monitoreo del día DD/MM/AAAA	53
Figura 21 Arquitectura Física.....	54
Figura 22 Arquitectura Lógica.....	55
Figura 23 Diagrama de Componentes	56
Figura 24 Implementación de Sensor de pH	60
Figura 25 Sensor de Temperatura.....	60
Figura 26 Implementación de Sensor Turbidez.....	61
Figura 27 Implementación de Modulo WIFI 8266 ESP-12	61
Figura 28 Sección donde se implementara el dispositivo	64
Figura 29 Pozo de agua	64
Figura 30 Sección donde se implementara el dispositivo y pozo de agua	64
Figura 31 Implementación Código para medir Temperatura.....	65
Figura 32 Implementación Código para medir pH	66
Figura 33 Implementación del código para medir la Turbidez.....	66
Figura 34 Configuración de módulo de wifi para el envío de información	67
Figura 35 Código para envío de información obtenida por sensores	67
Figura 36 Clase Entidad	68
Figura 37 Método para Listar Resultados de Lectura de los sensores.....	69
Figura 38 Método Insertar Resultados de lectura de sensores.....	69
Figura 39 Método Listar.....	70
Figura 40 Método para insertar datos	70
Figura 41 Código HTML del Menú Principal.....	71
Figura 42 Menú Principal.....	71
Figura 43 Código HTML de la Lista de Resultados	72
Figura 44 Código TypeScript de la Lista de Resultados	72
Figura 45 Lista de Resultados de los parámetros de calidad del agua	73
Figura 46 Datos de Monitoreo de parámetros de calidad del agua.....	74
Figura 47 Información de la Finalidad de la Aplicación.....	74
Figura 48 Gráfico de aceptación de Nivel de Completitud Funcional	82
Figura 49 Gráfico de Aceptación de Nivel de Corrección Funcional	86
Figura 50 Gráfico de Aceptación de Nivel de Comportamiento Temporal.....	91
Figura 51 Nivel de Completitud Funcional.....	92
Figura 52 Nivel de Corrección Funcional	93
Figura 53 Nivel de Comportamiento Temporal	94

RESUMEN

La presente investigación y propuesta tecnológica, tiene como principal objetivo implementar una aplicación móvil que permite monitorear la calidad de agua potable en pozos en los centros poblados del distrito de Laredo, Trujillo, midiéndose a través de un dispositivo electrónico evaluando los siguientes parámetros: Potencial de Hidrógeno (pH), la Temperatura y la Turbidez, utilizando para ello como tecnología Arduino mega y sensores adecuados para los parámetros antes mencionados. Esto nace desde la problemática que se presenta actualmente en nuestro país, donde los centros poblados que tienen el servicio de agua potable no se miden con frecuencia la calidad de agua, esto por el tiempo que demora para la recolección de datos, donde personal capacitado tiene que viajar a los centros poblados cada cierto tiempo, tomar las muestras correspondientes para luego analizarlas. De esta manera no se garantiza que el agua sea totalmente potable los 365 días del año. Dado esto la población suele estar muy susceptible a enfermarse al consumir esta agua sin hervir previamente. “En la selva y área rural existe un menor porcentaje de hogares con acceso de agua segura.” INEI (2016). Encuesta demográfica y de salud familiar.

Este trabajo se sustenta bajo el desarrollo de la metodología SCRUM, una metodología ágil que nos ayuda en el proceso del proyecto, ya que el desarrollo es flexible, productivo y trabajado en equipo. Así mismo el uso de la tecnología del Arduino mega en el proceso de medición para la calidad del agua. Los sensores a utilizar son los siguiente: Sensor PH, sensor de temperatura y sensor de turbidez. Posteriormente los datos obtenidos se almacenan en un servidor.

Obteniendo como resultado, una aplicación móvil que permite monitorear el nivel de la calidad del agua potable para los centros poblados del distrito de Laredo, Trujillo.

ABSTRACT

The present research and technological proposal, allows the monitoring of drinking water quality in wells of population centers of Laredo, Trujillo, measured through an electronic device, evaluating the following parameters: Potential of Hydrogen (pH), Temperature and Turbidity, using the Arduino mega technology and the appropriate parameters for the aforementioned parameters. This is born from the problem that is currently present in our country, where the population centers have drinking water service, water quality is not often measured, this is due to the time it takes to collect data, where trained personnel have to travel to populated centers every so often, take the corresponding samples and then analyze them. In this way it is not guaranteed that the water will be potable 365 days a year. Since the population is usually very susceptible to get sick by consuming this water without boiling previously. "In the jungle and the rural area there is a lower percentage of households with safe water access." INEI (2016). Demographic and family health survey.

This work is based on the development of the SCRUM methodology, an agile methodology that helps in the project process, since it is flexible, productive and worked as a team. So the use of the Arduino mega technology in the process of measuring for water quality. The sensors to be used are the following: PH sensor, temperature sensor and turbidity sensor. Subsequently, the data obtained is stored on a server.

Obtaining as a result, a mobile application that allows to monitor the level of drinking water quality for the population centers of Laredo, Trujillo.

CAPÍTULO 1. INTRODUCCIÓN

1.1. Realidad problemática

A lo largo de la historia de la humanidad, el agua es y seguirá siendo la bebida más importante para el ser humano, sin embargo, existen muchos países donde esta bebida aún no es saludable y de calidad para los seres humanos. Comúnmente esta bebida es consumida desde los grifos que se tiene en casa sin antes hervir; sin embargo, no estamos seguros de que el agua potable que consumimos sea de calidad. Siendo susceptibles a cualquier enfermedad por bacterias, parásitos o virus que ingerimos.

Un país no ajeno a esto es España, que presenta la misma problemática, siendo un país desarrollado. En la investigación de Montañez (2017), se afirma que: En la comparativa de la OCU, de las 62 muestras de agua recogidas en la red de hogares, presentaban problemas 7 de ellas, con deficiencias variadas: en Palma, el problema fue la presencia de nitratos; en Ciudad Real se detectaron trihalometanos; y el recuento de indicadores de contaminación fue alto en Huelva, Barcelona, Arándiga (Zaragoza), Logroño y Lebanza (Palencia). Los analistas encontraron aerobios y coliformes en las cuatro primeras localidades citadas (Palma, Ciudad Real, Huelva y Barcelona), y E. Coli en la última palentina, el caso «más preocupante» de cuantos analizó el laboratorio, pues puede causar problemas de salud.

Según Montañez (2017), los expertos señalan que en localidades pequeñas el agua suele estar peor controlada que en las grandes, esta aclaración es muy importante, puesto que, España siendo un país desarrollado presenta falencias con la potabilidad de agua en distintos lugares del país, y a raíz de esto podría causar problemas de salud.

En Colombia, presenta también esta problemática, es decir, las ciudades principales de este país cuentan con una mejor calidad de agua, que las zonas rurales o alejadas de la ciudad, siendo Barranquilla una de las ciudades con mejor calidad de agua, y Tamalameque una ciudad con la peor calidad de agua, teniendo el peor índice de calidad del agua en Colombia, según un estudio del Instituto Nacional de Salud (INS) del año 2015.

Entonces, podemos decir que las ciudades alejadas o centros poblados de este país, son los que no tienen evaluaciones frecuentes sobre la calidad de agua, y su población es las más afectada al contraer enfermedades producto de algún virus y/o bacteria. Y esto lo podemos observar en la investigación realizada por el Instituto Nacional de Salud (como se citó en el diario El Tiempo, 2016) mencionando lo siguiente:

Los municipios que tiene un índice entre 80,1 y 100 % de IRCA están en mayor riesgo debido a que el agua es inviable sanitariamente, como es el caso de Tamalemeque, que tiene 100. Por tanto, corren el riesgo de contraer enfermedades, tales como hepatitis A, cólera, fiebre tifoidea y en gran medida enfermedades diarreicas agudas. Todas consideradas de salud pública.

Sin embargo, esta problemática no solo se presenta en los países antes mencionados, sino también en el Perú, debido a que la calidad de agua potable no está garantizada a nivel nacional, solo las principales ciudades del país, como Lima tienen agua potable saludable. Según una encuesta realizada por Instituto Integración en 19 departamentos, indica que el 30% de la población que recibe agua potable es poco o nada saludable. Hildegardi Venero, investigadora del Instituto de Estudios Peruanos comentó, “Si en el ámbito urbano tenemos 55 % de la población que consume agua de calidad, en la zona rural tenemos 1.5 %. Mientras más cercano esté a Lima, recibe mejor agua de calidad”. Tomando en cuenta este último punto, en la zona rural y centros poblados son los más afectados por la calidad de agua potable, esto sucede en función al escaso monitoreo que se les hace a los pozos de agua, sin considerar el nivel de contaminación que pueden alcanzar en cortos plazos; por lo que, en nuestro país es altamente vulnerable a cambios climáticos.

Entonces, se puede decir que las zonas rurales son las más afectadas por este problema, no ajeno a esto ocurre con los centros poblados del distrito de Laredo, en Trujillo, La libertad. La cual, se realizaron diagnósticos elaborados por la comuna de Laredo en convenio con la Cooperación al Desarrollo Alemana G&Z, obteniendo como resultado lo siguiente: La zona rural, el 66% de los reservorios de abastecimiento de agua están deteriorados y el 80% de estos sistemas cuentan con cloración por difusión, pero actualmente su rendimiento no es duradero por los 10 años de antigüedad que tiene hasta el momento. (Anónimo, 2017)

Así mismo en una investigación realizada por los alumnos de la carrera de Ingeniería Ambiental de la Universidad César Vallejo (UCV), junto al microbiólogo Luis Cabanillas Chirinos, que, a través de dos sesiones, analizaron agua de caño, y en otra recogieron muestras del agua de un río, esto, para evaluar la calidad del agua de ambas muestras en el distrito de Laredo. La cual, menciona (como se citó en el diario Día a Día Trujillo, 2018): Los resultados obtenidos sobrepasan los límites máximo permisible según el Reglamento de calidad de Agua del Ministerio de Salud D.S. 031-2010-SA, tales valores obtenidos tienen la necesidad de realizar un tratamiento de desinfección para eliminar cualquier tipo de microorganismo causante a corto plazo, y poder evitar enfermedades diarreicas y digestivas.

Entonces, considerando el mal estado de los reservorios de agua, y el ineficiente control de calidad para dicho líquido en los reservorios, se puede deducir, que las personas que habitan en aquellos lugares tienen un riesgo alto a enfermarse por alguna bacteria que se ingiera al momento de beberla.

1.2. Formulación del problema

¿Es posible monitorear la calidad de agua potable, a través de una aplicación móvil complementada con la tecnología de sensores Multiparamétricos con Arduino?

1.3. Justificación

En esta investigación se propone desarrollar una aplicación móvil para monitorear la calidad de agua potable en las zonas rurales, la misma que será medida a través de un dispositivo tecnológico, que evalúa los parámetros: Potencial de Hidrogeno (pH), la Temperatura y la Turbidez; utilizando Arduino mega y sensores adecuados para dichos parámetros. Entonces, la importancia del proyecto, es garantizar que, en los centros poblados de Laredo se distribuya agua potable de calidad, que cumpla con los niveles adecuados de los parámetros mencionados, para así garantizar y cuidar la salud de las personas en los centros poblados, porque en la actualidad no existe un monitoreo adecuado que garantice frecuentemente la calidad de agua potable en estas zonas.

Nuestro objetivo va de la mano a lo estipulado por el Reglamento de la Calidad del Agua para Consumo Humano a lo largo de sus diez títulos. Dicho reglamento fue dado por la Dirección General de Salud Ambiental (DIGESA). A su vez, consideramos importante lo recomendado por la Nota Técnica de Prevención (NTP) N° 538, que se refiere a las medidas de prevención y control en instalaciones de suministro de agua, para evitar la proliferación de bacterias como la Legionela, a causa de temperaturas elevadas del agua potable en las instalaciones de suministro de agua.

Con ello la importancia de resaltar que nuestra investigación va a beneficiar principalmente a los pobladores, por lo que, al tener un monitoreo constante de los parámetros de calidad del agua, tienen la mayor certeza de estar consumiendo agua potable, esto con el fin de evitar posibles enfermedades por virus y/o bacterias que puedan contraer.

1.4. Limitaciones

- Limitación para la realización de las pruebas, ya que se necesita el permiso del municipio de Laredo, para poder acceder a pozos de agua y hacer las mediciones necesarias y así determinar si el agua es potable.
- Limitación para acceder a la información de los estudios realizados por el municipio de Laredo respecto a la potabilización de los pozos de agua.

Estas limitaciones se superaron, al solicitar un permiso para poder realizar las pruebas y para poder contar con la información de los estudios que realiza el municipio de Laredo.

Es importante mencionar, que el presente trabajo de investigación una vez implementado se puede extender al resto de provincias de Trujillo y del Perú, específicamente donde no haya un control adecuado de la potabilización del agua.

1.5. Objetivos

1.5.1. Objetivo general

Implementar una aplicación móvil complementada con la tecnología de sensores Multiparamétricos con Arduino, para monitorear la calidad del agua potable.

1.5.2. Objetivos específicos

- Lograr que la aplicación móvil alcance la completitud funcional, acorde a los umbrales determinados por la DIGESA.
- Lograr que la aplicación móvil provea resultados correctos con el nivel de precisión requerida, de acuerdo a los umbrales determinados por la DIGESA.
- Lograr que la aplicación móvil reduzca el tiempo de respuesta y procesamiento de los parámetros de calidad del agua.
- Establecer el nivel de eficacia de los resultados a través de un monitoreo discreto.

CAPÍTULO 2. MARCO TEÓRICO

2.1. Antecedentes

El-Khouri Vidarte, N. (2016). En su tesis titulada “Adaptación e Implementación de un Sistema Autónomo de Bajo Coste de Monitorización de Calidad del Agua en Tiempo Real” menciona que, en los países menos desarrollados, el acceso y control del agua es limitado. Por lo cual, según datos de la ONU mueren más de tres millones de personas, resaltando en ese sentido la prioridad de controlar la calidad del agua. En ese sentido, el principal objetivo de este autor fue diseñar e implementar un sistema de bajo de coste y autónomo para monitorizar la calidad del agua en tiempo real, usando para ello la placa Arduino Mega y sensores adecuados para el hardware antes mencionado, para medir los parámetros más significativos de la calidad del agua. Los parámetros escogidos fueron el Potencial de Hidrógeno (pH), Electro Conductividad (EC), la Temperatura y la Turbidez. Obteniendo como resultado el diseño y la implementación del sistema de bajo coste para el control de calidad de agua que permite abarcar alrededor del 90% de los casos de agua contaminada. Para ello, realizó pruebas que demostraron que el sistema alcanzo la funcionalidad esperada. Los resultados obtenidos en esta investigación, nos es de mucha importancia, puesto que, en nuestra investigación medimos los mismos parámetros (potencial de hidrógeno, temperatura y turbidez) para la calidad de agua, para la cual nos ayuda a comparar en la obtención de datos y los resultados generados.

Román Herrera, C., Loza Matovelle, D., Segura, L. y Dabirian, R (2015). En su paper titulado “Construcción con tecnología abierta de un sensor de turbidez de bajo costo” menciona, la importancia de controlar la turbidez en fuentes de agua superficial esencialmente en zonas rurales. Por ello la aplicabilidad de su proyecto la direccionan a las plantas de tratamiento de aguas servidas en las regiones rurales de Ecuador. Siendo su objetivo construir un sensor de turbidez de bajo costo, con tecnología abierta, como Arduino, impresoras 3D y software libre. El dispositivo consta de un LED que emite luz y dos sensores receptores dispuestos a 90°, que miden la luz que atraviesa el líquido. Además, se realizaron pruebas de calibración obteniendo un rango efectivo entre 50 NTU hasta 650 unidades nefelométrías de turbidez (NTU). Los resultados alcanzados fueron bastante alentadores pues para validar los resultados se comparó con turbidímetro comercial, obteniendo resultados con un error de 3.3%. Siendo entonces, esta investigación de suma importancia para nosotros, al considerar el sensor de turbidez parte de su tecnología, para la cual, nuestra investigación utilizará también dicho sensor.

Aguirre Obregón, M., Carrasco Peña, L. y Garay Quiñones, J. (2016). En su tesis titulada “Sistema Web para el Seguimiento y Control del Proceso de Parámetros de Calidad de Agua y Ambiente en

el Instituto del Mar del Perú (IMARPE) del Ministerio de Producción del Perú” menciona que, al ser las investigaciones de realizadas por parte del Instituto del Mar del Perú, de suma importancia comercial, deben siempre brindar datos de la más alta confiabilidad; sin embargo, observaron que la recolección de datos para controlar los parámetros de calidad de agua y ambiente eran manuales por lo cual estaban expuestas a alteraciones por deterioro o pérdida. En ese sentido, la investigación se centró en implementar un sistema web para el seguimiento y control del proceso de parámetros de calidad de agua y ambiente, optimizando el proceso de recolección de datos automáticamente para la medición de parámetros fisicoquímicos de tanques de cultivo de micro algas tanto marinas como continentales. Pretendiendo no solo automatizar la tarea sino también ahorrar tiempo en los procesos de medición. Usaron metodología RUP para el desarrollo e implementación del software, tecnología libre como el Arduino y los sensores adecuados (temperatura, pH, oxígeno disuelto y sensor de Luz), para el proceso de captura de datos de los parámetros de calidad de agua y ambiente. Concluyendo, que la implementación del sistema web resulta eficiente para un mejor control de los parámetros físicos químicos del agua y ambiente en IMARPE. Asimismo, demostraron que la interacción entre el sistema web y el Arduino mejorará el proceso de captura de datos y permitirá el registro automático de los datos. Por último, determinaron a través de un análisis del valor actual neto (VAN), el que resulto mayor que cero, garantiza la rentabilidad económica de los investigadores. En esta investigación, se demuestra que la interacción entre el sistema web y el Arduino, mejora el proceso en la captura de datos; esto nos ayudará a comparar dicho proceso, por lo que nuestra investigación se relaciona aplicación móvil con Arduino para la captura de datos.

Cava Suárez, T., Ramos Arévalo, F. (2016). En su tesis “Caracterización físico – química y microbiológica de agua para consumo humano de la localidad Las Juntas del distrito Pacora – Lambayeque, y propuesta de tratamiento”, menciona que para el análisis de agua se tomaron diez puntos de muestreo en diferentes sitios de la localidad los cuales incluye el pozo subterráneo, tanque de almacenamiento y 8 viviendas, para cada sitio de muestreo se recolectó dos muestras para análisis físico – químico y microbiológico respectivamente, se recolectó por 4 semanas haciendo un total de 40 muestras, evaluando parámetros físicos, químicos y bacteriológicos. Obteniéndose como resultado, que está dentro de los límites para consumo humano en: pH, dureza total, turbidez, color, nitratos, arsénico, plomo y recuento de heterótrofos. Mientras que los siguientes parámetros sobrepasan los límites para consumo humano: cloruros entre 270 - 298 mg/L, magnesio entre 30,8 - 41,2 mg/L, conductividad eléctrica entre 3400 - 3475 $\mu\text{s}/\text{cm}$, solidos totales disueltos entre 2040 - 2085 mg/L, sulfatos entre 455,2 - 490,2 mg/L, cloro residual con 0 ppm, coliformes totales entre 30 - 50 UFC/100ml y coliformes termo tolerantes entre 1 - 2 UFC/100ml, por lo que puede afectar la salud del consumidor. Así mismo, los autores recomiendan implementar un programa de vigilancia y monitoreo de la calidad de agua de Las Juntas del distrito de Pacora, con el fin de verificar y garantizar si el agua es apta para el consumo humano. En esta investigación indica que los datos obtenidos están dentro de los límites para el consumo humano, siendo esto una investigación muy

eficiente con poco margen de error. Para la cual, será nuestra base a comparar con el parámetro, potencial de hidrógeno.

2.2. Bases teóricas

2.2.1. Parámetros de Calidad del Agua

De acuerdo al conocimiento de la existencia de compuestos en el agua potable, que pueden ser inocuos o dañinos para la población, se determinan los límites máximos permisibles de los parámetros de calidad del agua.

Figura 1 Límites Máximos permisibles referenciales de los parámetros de calidad del agua potable

Parámetro	LMP ref.	Referencias
Coliformes totales, UFC/100 mL	0 (ausencia)	1
Coliformes termotolerantes, UFC/100 mL	0 (ausencia)	1
Bacterias heterotróficas, UFC/mL	500	1
pH	6,5-8,5	1
Turbiedad, UNT	5	1
Conductividad, 25 °C IS/cm	1.500	3
Color, UCV-Pt-Co	20	2
Cloruros, mg/L	250	2
Sulfatos, mg/L	250	2
Dureza, mg/L	500	3
Nitratos, mg NO ₃ /L	50	1
Hierro, mg/L	0,3	0,3 (Fe + Mn = 0,5) 2
Manganeso, mg/L	0,2	0,2 (Fe + Mn = 0,5) 2
Aluminio, mg/L	0,2	1
Cobre, mg/L	3	2
Plomo, mg/L	0,1	2
Cadmio, mg/L	0,003	1
Arsénico, mg/L	0,1	2
Mercurio, mg/L	0,001	1
Cromo, mg/L	0,05	1
Flúor, mg/L	2	2
Selenio, mg/L	0,05	2

Fuente: [SUNASS], 2004

2.2.1.1. Temperatura

Según ([Digesa], 2006), la temperatura del agua es un parámetro importante sobre el desarrollo de la vida acuática, reacciones químicas y velocidades de reacción, así como la aptitud del agua para ciertos usos útiles. La temperatura es un indicador de la calidad del agua, que influye en el comportamiento de otros indicadores de la calidad del recurso hídrico, como el pH, el déficit de oxígeno, la conductividad eléctrica y otras variables fisicoquímicas.

La Organización Mundial de la salud ([OMS], 2006) respecto a la temperatura, menciona que las bacterias como las legionelas, pueden proliferarse a temperaturas elevadas existentes en ocasiones en los sistemas de distribución de agua de consumo entubada, por ello la importancia de monitorear la temperatura del agua.

A su vez, el ministerio de trabajo y asuntos sociales de España ([MITRAMISS], 1999), comenta en su NTP (guía de buenas prácticas) sobre medidas de prevención y control en instalaciones de suministro de agua, que dicha bacteria (Legionela) se desarrolla en agua a temperaturas oscilen entre 20 °C y 50 °C, con un desarrollo óptimo entre 35 °C y 45 °C. Por debajo de los 20 °C permanece latente, sin multiplicarse, y no sobrevive por encima de los 60 °C.

Características

De acuerdo con ([DIGESA], 2006), las características de la temperatura son:

- El oxígeno es menos soluble en agua caliente que en agua fría.
- Un cambio brusco de temperatura puede conducir a un aumento en la mortalidad de la vida acuática.
- La temperatura óptima para el desarrollo de las actividades se detienen cuando se alcanza los 50°C, a temperaturas de alrededor de 15°C, las bacterias productoras de metano cesan su actividad.

Riesgos

Como afirma ([DIGESA], 2006), los riesgos que se pueden derivar de los cambios anormales en la temperatura del agua son:

- Las temperaturas anormalmente elevadas pueden dar lugar a una indeseada proliferación de plantas acuáticas y hongos.

- En periodos extendido de continua inmersión en agua más fría o $<15^{\circ}\text{C}$ puede causar la muerte de algunos bañistas y será riesgo para todos los bañistas que no usen ropa protectora de inmersión. La sobrevivencia de un individuo sumergido en agua por encima de 34 o 35°C va depender de la tolerancia a una elevada temperatura corporal interna, a un riesgo de daño con la exposición prolongada.

Método de Análisis

De acuerdo a ([DIGESA], 2006), este parámetro es medido in situ: el método termometría.

2.2.1.2. Potencial de Hidrógeno (pH)

Según ([DIGESA], 2006), el pH es el valor que determina si una sustancia es ácida, neutra o básica, calculando el número iones hidrogeno presentes. Se mide en una escala a partir de 0 a 14, en la escala 7, la sustancia es neutra. Los valores de pH por debajo de 7 indican que una sustancia es ácida y los valores de pH por encima de 7 indican que es básica. Cuando una sustancia es neutra el número de los átomos de hidrógeno y de oxhidrilos son iguales. Cuando el número de átomos de hidrógeno (H^+) excede el número de átomos del oxhidrilo (OH^-), la sustancia es ácida.

Características

Conforme a ([DIGESA], 2006), las características del pH son:

- La concentración de ion hidrógeno es un parámetro de calidad de gran importancia tanto para el caso de calidad de las aguas naturales como residuales.
- Todas las fases del tratamiento del agua para suministro y residual, como o la neutralización ácida – base, suavizado, precipitación, coagulación, desinfección y control de la corrosión, depende del pH.
- El agua residual con concentración de ion hidrógeno presenta elevadas dificultades de tratamiento con procesos biológicos y el efluente puede modificar la concentración de ion hidrógeno en las aguas naturales si ésta no se modifica antes de la evacuación de las aguas.
- A una temperatura determinadas, la intensidad del carácter ácido o básico de una solución viene dada por la actividad del ion hidrogeno o pH.
- El pH de los sistemas acuosos puede medirse convenientemente con pH-metro.

Riesgos

Tal como afirma ([DIGESA], 2006), los riesgos que ocasiona el consumir agua con pH elevado al recomendado, son los siguientes:

- El pH no ejerce efectos directos en los consumidores, es uno de los parámetros indicadores de la calidad del agua. Para que la desinfección con cloro sea eficaz es preferible que sea un pH inferior a 8.
- En valores superiores de pH 11 produce irritación ocular y agravación de trastornos cutáneos.

Método de Análisis

Conforme con ([DIGESA], 2006), es recomendable la medición in situ, de modo que no se modifique los equilibrios iónicos. debido al transporte o una permanencia prolongada en recipientes cambia cuando es llevado al laboratorio, el método aplicado en in situ es método electrométrico.

Figura 2 Escala del pH.



Fuente: Carbotecnia, 2014

2.2.1.3. Electro Conductividad

Según ([DIGESA], 2006), depende de la actividad de los tipos de iones disueltos y de la temperatura a la que se realiza la medida. La conductividad es una expresión numérica de la capacidad de una solución para transportar una corriente eléctrica. Esta capacidad depende de la presencia de iones y de su concentración total, de su movilidad, valencia y concentraciones relativas, así como la temperatura de la medición. El agua pura tiene muy poca conductividad, por lo que la medida de la conductividad de un agua nos da una idea de los sólidos disueltos en la misma. De la conductividad eléctrica, que indica la presencia de sales en el agua, lo que hace aumentar su capacidad de transmitir una corriente eléctrica, propiedad que se utiliza

en mediciones de campo o de laboratorio, expresadas en micro Siemens/l ($\mu\text{S/l}$). A partir de la conductividad se puede obtener los sólidos disueltos multiplicando por un factor entre 0.55 y 0.75. Los sólidos disueltos totales, expresados en mg/L, pueden ser obtenidos por multiplicación de la conductividad por un factor comprendido entre 0,55 y 0,75. Este factor puede ser determinado para cada cuerpo de agua, pero permanece aproximadamente constante, según las proporciones iónicas en el cuerpo de agua y si éstas permanecen estables.

Características:

De acuerdo a ([DIGESA], 2006), las características del electro conductividad en las soluciones son:

- Las soluciones de la mayoría de ácidos, bases y sales presentan coeficientes de conductividad relativamente adecuados.
- Las moléculas de los compuestos orgánicos que no se disocian en soluciones acuosas tienen una conductividad muy escasa o nula.
- La conductividad eléctrica de un agua se utiliza como una medida indirecta de su concentración de sólidos disueltos totales o de minerales en el agua.
- La salinidad del agua se determina midiendo su conductividad eléctrica. La presencia de sales afecta el crecimiento de las plantas por tres mecanismos.

1. Efectos osmóticos, provocados por la concentración total de sales en el agua del suelo.
2. Toxicidad de iones específicos, provocada por la concentración de un ion determinado.
3. Dispersión de las partículas de suelo, provocada por la presencia importante de sodio y por una baja salinidad. Es habitual encontrar valores de 700 $\mu\text{mhos/cm}$. a 1200 $\mu\text{mhos/cm}$ de manera natural en cuerpos de agua superficiales. La conductividad eléctrica se expresa en $\mu\text{hm/cm}$.

2.2.1.4. Turbidez

Conforme a ([DIGESA], 2006), la turbidez del agua es producida por materias en suspensión, como arcillas, cieno o materias orgánicas e inorgánicas finamente divididas, compuestos orgánicos solubles coloreados, plancton, sedimentos procedentes de la erosión y microorganismos, el tamaño de estas partículas varía desde 0,1 a 1.000 nm (nanómetros) de diámetro. La turbidez se utiliza para indicar la calidad del agua y la eficiencia de la filtración para determinar si hay presencia de

organismos que provocan enfermedades. La materia suspendida en el agua absorbe la luz, haciendo que el agua tenga un aspecto nublado. La turbidez se puede medir con varias diversas técnicas, esto demuestra la resistencia a la transmisión de la luz en el agua.

Características:

De acuerdo a ([DIGESA], 2006), las características de la turbidez son:

- La turbiedad, como medida de las propiedades de transmisión de la luz de un agua, es otro parámetro que se emplea para indicar la calidad de las aguas vertidas o de las aguas naturales en relación con la materia coloidal y residual en suspensión.
- Elevados niveles de turbiedad pueden proteger a los microorganismos de los efectos de la desinfección y estimular la proliferación de bacteria.

Riesgos

Conforme a ([DIGESA], 2006), los riesgos de consumir agua con un nivel de turbidez elevado son:

- Una alta turbidez suele asociarse a altos niveles de microorganismos como virus, parásitos y algunas bacterias. Estos organismos pueden provocar síntomas tales como nauseas, retortijones, diarreas y dolores de cabeza.

Métodos de Análisis

Método Nefelométrico son expresados en UNT (Unidades nefelométricas de Turbidez)

2.2.2. Arduino

Arduino (Arduino, 2018), es una plataforma electrónica de código abierto basada en hardware y software fáciles de usar. Las placas Arduino pueden leer entradas (luz en un sensor, un dedo en un botón o un mensaje de Twitter) y convertirla en una salida: activar un motor, encender un LED y publicar algo en línea. Puede decirle a su tarjeta qué debe hacer enviando un conjunto de instrucciones al microcontrolador en la tarjeta. Para hacerlo, utiliza el lenguaje de programación Arduino (basado en Wiring) y el software Arduino (IDE), basado en el procesamiento.

A lo largo de los años, Arduino ha sido el cerebro de miles de proyectos, desde objetos cotidianos hasta instrumentos científicos complejos. Una comunidad mundial de creadores

(estudiantes, aficionados, artistas, programadores y profesionales) se ha reunido en torno a esta plataforma de código abierto, sus contribuciones han sumado una cantidad increíble de conocimientos accesibles que pueden ser de gran ayuda para principiantes y expertos por igual.

Arduino nació en el Instituto de Diseño de Interacción (IVREA) como una herramienta fácil para la creación rápida de prototipos, dirigida a estudiantes sin experiencia en electrónica y programación. Tan pronto como llegó a una comunidad más amplia, la placa Arduino comenzó a cambiar para adaptarse a las nuevas necesidades y desafíos, diferenciando su oferta de tablas simples de 8 bits a productos para aplicaciones IoT, impresión portátil, impresión 3D y entornos integrados. Todas las placas Arduino son completamente de código abierto, lo que permite a los usuarios construirlas de forma independiente y, eventualmente, adaptarlas a sus necesidades particulares. El software también es de código abierto y está creciendo a través de las contribuciones de los usuarios de todo el mundo.

Arduino tiene una amplia gama completa de productos oficiales, como tarjetas de desarrollo (BOARDS), módulos (SHIELDS) y kits de desarrollo.

Figura 3 Comparación de las especificaciones de tarjetas de desarrollo

Name	Processor	Operating/Input Voltage	CPU Speed	Analog In/Out	Digital IO/PWM	EEPROM [kB]	SRAM [kB]	Flash [kB]	USB	UART
Mega 2560	ATmega2560	5 V / 7-12 V	16 MHz	16/0	54/15	4	8	256	Regular	4
Uno	ATmega328P	5 V / 7-12 V	16 MHz	6/0	14/6	1	2	32	Regular	1
Nano	ATmega168	5 V / 7-9 V	16 MHz	8/0	14/6	0.512	1	16	Mini	1
	ATmega328P					1	2	32		

Fuente: Arduino, 2018

Software Arduino (IDE)

El entorno de desarrollo integrado de Arduino (Arduino, 2018), o el software Arduino (IDE), contiene un editor de texto para escribir código, un área de mensajes, una consola de texto, una barra de herramientas con funciones comunes y una serie de menús. Se conecta al hardware Arduino y Genuino para cargar programas y comunicarse con ellos.

Los programas escritos utilizando el software Arduino (IDE) se denominan bocetos (sketches). Estos bocetos se escriben en el editor de texto y se guardan con la extensión de archivo “.ino”. El editor tiene características para cortar / pegar y para buscar / reemplazar texto. El área de mensajes proporciona comentarios al guardar y exportar y también muestra errores.

La consola muestra la salida de texto del software Arduino (IDE), que incluye mensajes de error completos y otra información. La esquina inferior derecha de la ventana muestra la placa configurada y el puerto serie. Los botones de la barra de herramientas le permiten verificar y cargar programas, crear, abrir y guardar bocetos, y abrir el monitor serie.

Le permite administrar bocetos con más de un archivo. Estos pueden ser archivos de código Arduino normales (sin extensión visible), archivos C (extensión .c), archivos C ++ (.cpp) o archivos de encabezado (.h).

2.2.3. IONIC (Mobile App Framework)

Ionic Framework (Ionic Framework, 2015) es un SDK de código abierto que permite a los desarrolladores crear aplicaciones móviles de gran calidad y rendimiento utilizando tecnologías web conocidas (HTML, CSS y JavaScript). Se centra principalmente en la apariencia y la interacción de la interfaz de usuario de una aplicación, simplificando una gran parte de su proceso de desarrollo de aplicaciones: el front-end. Adicionalmente, es completamente gratuito y de código abierto, publicado bajo la licencia de MIT, lo que significa que puede utilizar en proyectos personales o comerciales de forma gratuita.

El uso de Ionic en empresas puede tener diversas ventajas y desventajas (Esaú A., 2016), como las que menciono a continuación:

Ventajas:

- Desde una única fuente podremos llegar a las plataformas que soporta este framework (Android e iOS).
- El desarrollo principal se realiza en HTML junto con CSS y JS, lenguajes muy extendidos por la comunidad de desarrolladores, con lo que la implantación de esta herramienta en la empresa, facilitará el desarrollo de proyectos de la forma más efectiva aun cuando la plantilla de desarrolladores contenga nuevas incorporaciones.
- Si ya contamos con una web app que queremos convertir en aplicación móvil, en la mayoría de los casos habremos hecho uso de JavaScript, por lo que el código es reutilizable.
- Para el caso de aplicaciones híbridas, tendremos con un único proceso de desarrollo e implementación, una app para Android, iOS y web.

Desventajas:

- El rendimiento puede ser ligeramente menor que en aplicaciones desarrolladas de forma nativa, cosa que no debería ser un problema al menos que el proyecto sea para la creación de juegos con detallados gráficos u otras aplicaciones que hagan uso de grandes cantidades de recursos.
- Es una herramienta “joven” y puede ser difícil encontrar módulos compartidos por los usuarios, pero como dijimos, la comunidad está creciendo a pasos agigantados y en breve este inconveniente podría dejar de existir.
- Debido a esta “juventud” del framework, Ionic siguen cambiando y afinando algunas características tanto del framework como de sus normas de soporte, bibliotecas y demás se refiere; haciendo por tanto que los usuarios se actualicen cada cierto tiempo.

2.2.4. Angular JS

Según Azaustre C. (2013) indica que, es un framework MVC de JavaScript para el Desarrollo Web Front End que permite crear aplicaciones SPA *Single-Page Applications*. Entra dentro de la familia de frameworks como BackboneJS o EmberJS..

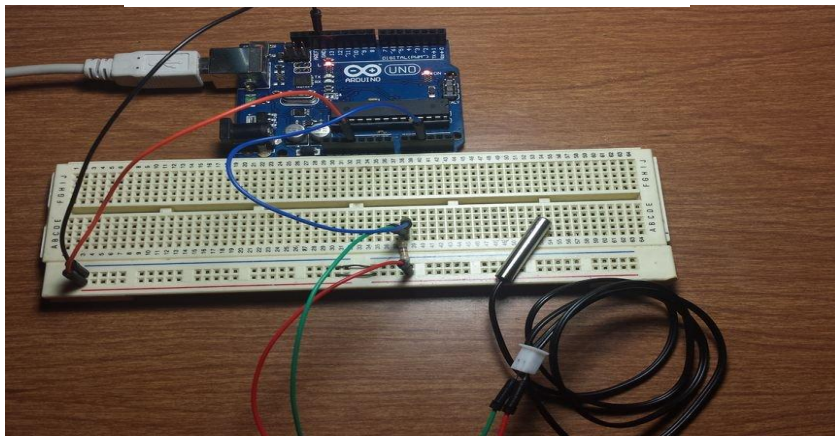
AngularJS permite extender el vocabulario HTML con directivas y atributos, manteniendo la semántica y sin necesidad de emplear librerías externas como jQuery Underscore.js para que funcione

2.2.5. Sensores

2.2.5.1. Sensor de Temperatura NTC 3950

De acuerdo a lo afirmado por (Electrocrea, 2014), este sensor es un simple termistor encapsulado en un tubo de acero a prueba de agua y corrosión. Útil para cuando necesite medir algo lejano, o en condiciones de humedad. El sensor alcanza hasta 125 °C, el cable está revestido en PVC por lo que se sugiere mantenerla por debajo de 100 °C. Debido a que son digitales, no recibe ningún tipo de degradación de la señal incluso en largas distancias. Compatible con sistemas de 3.0 ó 5.5V.

Figura 4 Sensor de Temperatura NTC 3950



Fuente: Electrocrea, 2014

2.2.5.2. Sensor de Turbidez SEN0189

Como afirma (DFRobot, 2017), el sensor de turbidez detecta la calidad del agua al medir los niveles de turbidez. Utiliza la luz para detectar partículas suspendidas en el agua al medir la transmitancia de la luz y la velocidad de dispersión, que cambia con la cantidad de sólidos suspendidos totales (SST) en el agua. Los sensores de turbidez se utilizan para medir la calidad del agua en ríos y arroyos, mediciones de aguas residuales y efluentes, instrumentación de control para estanques de sedimentación, investigación de transporte de sedimentos y mediciones de laboratorio.

Figura 5 Sensor de Turbidez SEN0189



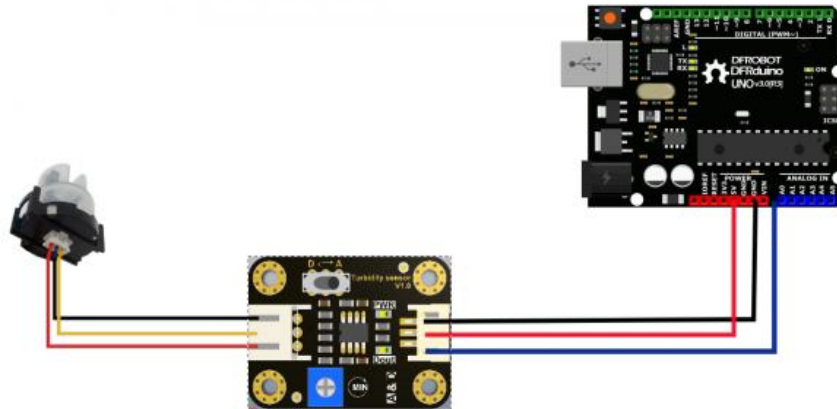
Fuente: DFRobot, 2017

Especificación

- Voltaje de funcionamiento: 5V DC
- Corriente de funcionamiento: 40mA (MAX)
- Tiempo de respuesta: <500ms
- Resistencia de aislamiento: 100M (Min)
- Método de salida:
- Salida analógica: 0-4.5V
- Salida digital: señal de nivel alto / bajo (puede ajustar el valor de umbral ajustando el potenciómetro)
- Temperatura de funcionamiento: 5 °C ~ 90 °C
- Temperatura de almacenamiento: -10 °C ~ 90 °C
- Peso: 30 g
- Dimensiones del adaptador: 38 mm * 28 mm * 10 mm / 1.5 pulgadas * 1.1 pulgadas * 0.4 pulgadas

Diagrama de Conexión

Figura 6 Diagrama de Conexión de Sensor de Turbidez al Arduino



Fuente: DFRobot, 2017

2.2.5.3. Sensor de Conductividad Eléctrica DFR0300

Como menciona (DFRobot, 2017), el Sensor de conductividad eléctrica mide la cantidad de corriente eléctrica que un material puede transportar. Por ejemplo, la salinidad de los suelos, sistemas de agua de riego o soluciones de fertilizantes. La conductividad del agua puede reflejar el nivel de electrolitos presentes en el agua. Dependiendo de la concentración de dicho elemento. Está diseñado especialmente para los controladores de Arduino y se ha incorporado en características simples, convenientes y prácticos.

Figura 7 Sensor de Conductividad Eléctrica DFR0300.



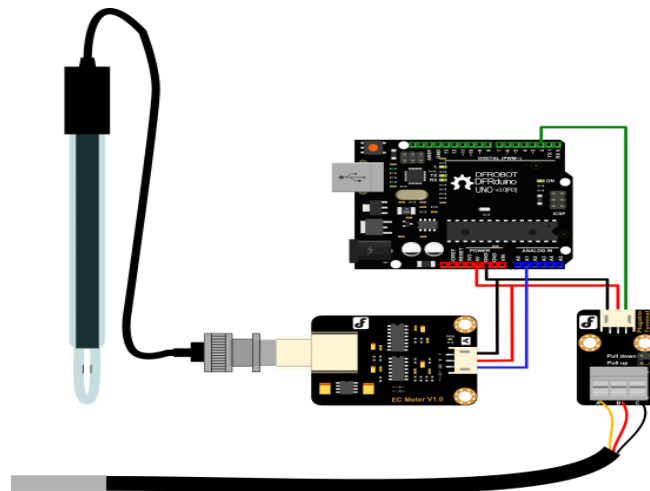
Fuente: DFRobot, 2017

Especificación

- Tensión de funcionamiento: +5.00 V
- Tamaño de PCB: 45mm x 32mm
- Rango de medición: 1 ms / cm - 20 ms / cm
- Temperatura de funcionamiento: 5-40
- Precisión: $\pm 10\%$ FS (la precisión específica depende de la precisión de su solución de calibración)
- Interfaz PH2.0 (SMD de 3 pines)
- Electrodo de conductividad (electrodo constante K = 1, conector BNC)
- Longitud del cable del electrodo: unos 60 cm
- Sensor de temperatura DS18B20 (impermeable)
- Indicador de encendido

Diagrama de Conexión

Figura 8 Diagrama de Conexión de Sensor de Conductividad Eléctrica DFR0300 al Arduino.



Fuente: DFRobot, 2017

2.2.5.4. Sensor de Ph SEN0161

Como afirma (DFRobot, 2017), el pH es una medida de acidez o alcalinidad de una disolución. El pH indica la concentración de iones hidronio $[H_3O^+]$ presentes en determinadas sustancias. Este kit permite medir de forma sencilla el pH de un líquido gracias a su placa controladora que ofrece un valor analógico proporcional a la medición. El controlador tiene un potenciómetro multivuelta que permite la correcta calibración de la sonda.

Figura 9 Sensor de pH SEN0161.



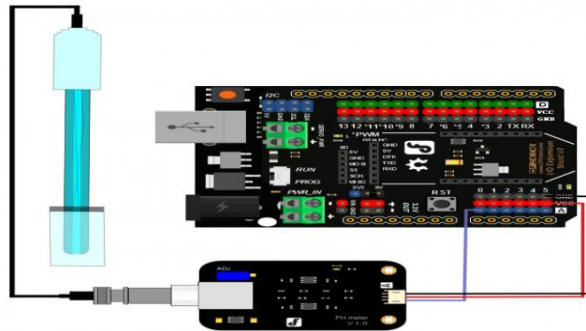
Fuente: DFRobot, 2017

Especificación

- Alimentación: 5.00V
- Dimensiones: 43x32mm (controlador)
- Rango de medición: 0-14 pH
- Temperatura de medición: 0-60 °C
- Precisión: ± 0.1 pH (25 °C)
- Tiempo de respuesta: ≤ 1 min
- Sonda de pH con conector BNC
- Controlador pH 2.0 (3 pines)
- Ajuste de ganancia
- Indicador LED

Diagrama de Conexión

Figura 10 Diagrama de Conexión de Sensor de pH SEN0161 al Arduino



Fuente: DFRobot, 2017

2.2.6. Metodologías de Desarrollo

Según (Jacobson I, 1999) una metodología de desarrollo de software en ingeniería de software es un marco de trabajo usado para estructurar, planificar y controlar el proceso de desarrollo en sistemas de información.

Dentro de las cuales actualmente tenemos las metodologías tradicionales y las metodologías ágiles, que son las que mayor apogeo tienen actualmente.

2.2.6.1. Metodologías Ágiles

En concordancia con (Cockburn A., 2001), el desarrollo ágil envuelve un enfoque para la toma de decisiones en los proyectos de software, que se refiere a métodos de ingeniería del software basados en el desarrollo iterativo e incremental, donde los requisitos y soluciones evolucionan con el tiempo según la necesidad del proyecto. Así el trabajo es realizado mediante la colaboración de equipos auto-organizados y multidisciplinarios, inmersos en un proceso compartido de toma de decisiones a corto plazo.

Figura 11 Desarrollo Tradicional vs. Desarrollo Ágil



Fuente: Malpica C., 2014.

2.2.5.1.1. Programación Extrema (XP)

De acuerdo a lo afirmado por (Ecured, 2018), la Programación Extrema PX, mejor conocida por su nombre en inglés Extreme Programming (PX), nace como nueva disciplina de desarrollo de software hace aproximadamente unos seis años, y ha causado un gran revuelo entre el colectivo de programadores del mundo. Kent Beck, su autor, es un programador que ha trabajado en múltiples empresas y que actualmente lo hace como Programador en la conocida empresa automovilística DaimlerChrysler. La programación extrema se basa en la simplicidad, la comunicación y el reciclado continuo de código. Los Valores originales de la programación extrema son: simplicidad, comunicación, retroalimentación (feedback) y coraje. Un quinto valor, respeto, fue añadido en la segunda edición de Extreme Programming Explained.

Las Cuatro Actividades Básicas

Codificar: Es la única actividad de la que no podremos prescindir. Por tanto, necesitamos codificar y plasmar nuestras ideas a través del código. En una programación en PX en pareja el código expresa tu interpretación del problema, así podemos utilizarlo para comunicar, para hacer más tus ideas, y por tanto para aprender y mejorar.

Hacer pruebas: Las características del software que no pueden ser demostradas mediante pruebas simplemente no existen. Las pruebas nos indican que nuestro trabajo funciona.

Escuchar: Los programadores no lo conocemos todo, y sobre todo muchas cosas que las personas de negocios piensan que son interesantes. Si vamos a hacer pruebas tenemos que preguntar si lo obtenido es lo deseado, y tenemos que preguntar a quién necesita la información.

Diseñar: El Diseño crea una estructura que organiza la lógica del sistema, un buen diseño permite que el sistema crezca con cambios en un solo lugar. Los diseños deben de ser sencillos, si alguna parte del sistema es de desarrollo complejo, divídela en varias.

2.2.6.2. SCRUM

Como afirma (Ecured, 2018), es un proceso en el que se aplican de manera regular un conjunto de buenas prácticas para trabajar colaborativamente, en equipo, y obtener el mejor resultado posible de un proyecto. Estas prácticas se apoyan unas a otras y su selección tiene origen en un estudio de la manera de trabajar de equipos altamente productivos. En Scrum se realizan entregas parciales y regulares del producto final, priorizadas por el beneficio que aportan al receptor del proyecto. Por ello, Scrum está especialmente indicado para proyectos en entornos complejos, donde se necesita obtener resultados pronto, donde los requisitos son cambiantes o poco definidos, donde la innovación, la competitividad, la flexibilidad y la productividad son fundamentales.

Actividades de SCRUM

Planificación de la iteración

El primer día de la iteración se realiza la reunión de planificación de la iteración tiene dos partes:

- **Selección de requisitos (4 horas máximo).** El cliente presenta al equipo la lista de requisitos priorizada del producto o proyecto. El equipo pregunta al cliente las dudas que surgen y selecciona los requisitos más prioritarios que se compromete a completar en la iteración, de manera que puedan ser entregados si el cliente lo solicita.

- **Planificación de la iteración (4 horas máximo).** El equipo elabora la lista de tareas de la iteración necesarias para desarrollar los requisitos a que se ha comprometido. La estimación de esfuerzo se hace de manera conjunta y los miembros del equipo se auto asignan las tareas.

Ejecución de la iteración

Cada día el equipo realiza una reunión de sincronización (15 minutos máximos). Cada miembro del equipo inspecciona el trabajo que el resto está realizando (dependencias entre tareas, progreso hacia el objetivo de la iteración, obstáculos que pueden impedir este objetivo) para poder hacer las adaptaciones necesarias que permitan cumplir con el compromiso adquirido. En la reunión cada miembro del equipo responde a tres preguntas:

- ¿Qué he hecho desde la última reunión de sincronización?
- ¿Qué voy a hacer a partir de este momento?
- ¿Qué impedimentos tengo o voy a tener?

Durante la iteración el Facilitador:

- Se encarga de que el equipo pueda cumplir con su compromiso y de que no se merme su productividad.
- Elimina los obstáculos que el equipo no puede resolver por sí mismo.
- Protege al equipo de interrupciones externas que puedan afectar su compromiso o su productividad.

Inspección y adaptación

El último día de la iteración se realiza la reunión de revisión de la iteración la cual tiene dos partes:

- **Demostración (4 horas máximo).** El equipo presenta al cliente los requisitos completados en la iteración, en forma de incremento de producto preparado para ser entregado con el mínimo esfuerzo. En función de los resultados mostrados y de los cambios que haya habido en el contexto del proyecto, el cliente realiza las adaptaciones necesarias de manera objetiva, ya desde la primera iteración, re planificando el proyecto.
- **Retrospectiva (4 horas máximo).** El equipo analiza cómo ha sido su manera de trabajar y cuáles son los problemas que podrían impedirle progresar

adecuadamente, mejorando de manera continua su productividad. El Facilitador se encargará de ir eliminando los obstáculos identificados.

2.2.5.2. Metodologías de desarrollo tradicionales

Las metodologías tradicionales se consideran estrictas, por este motivo se les suele denominar con el término de “pesadas”, ya que proponen la creación de una documentación exhaustiva, así como una planificación y seguimiento riguroso de las múltiples actividades a realizar durante el proyecto.

2.2.5.2.1. RUP

Como afirma (Jacobson I., 1999), el Proceso Unificado Racional (RUP), es un proceso de desarrollo de software desarrollado por la empresa Rational Software, actualmente propiedad de IBM. Junto con el Lenguaje Unificado de Modelado (UML), constituye la metodología estándar más utilizada para el análisis, diseño, implementación y documentación de sistemas orientados a objetos.

CAPÍTULO 3. HIPÓTESIS

2.3. Formulación de la Hipótesis

La implementación de una aplicación móvil, complementada con la tecnología de sensores Multiparamétricos con Arduino permite monitorear la calidad del agua potable.

2.4. Operacionalización de variables

Tabla 1 Operacionalización de variables

VARIABLE	DEFINICIÓN CONCEPTUAL	DIMENSIONES	INDICADORES
Variable Dependiente: Monitoreo de la calidad del agua potable	Se visualiza el resultado de la medición, tanto los valores obtenidos, como la calibración de la potabilidad del agua.	Eficacia	Nivel de eficacia de resultado
Variable Independiente: Aplicación móvil complementada con la tecnología de sensores Multiparamétricos con Arduino	Se obtendrán los datos de los parámetros pH, temperatura y turbidez, a través de los sensores correspondientes, que estarán conectados en el Arduino, donde éste procesará la información.	Funcionalidad	Complejidad funcional Corrección funcional
		Rendimiento	Comportamiento temporal

Fuente: Elaboración Propia

CAPÍTULO 4. DESARROLLO

4.1. Metodología

Se ha seleccionado SCRUM entre otras metodologías ágiles y tradicionales, basado en varios factores de carácter técnico y productivo, las metodologías analizadas son SCRUM y RUP. La siguiente tabla comparativa nos muestra que, de las dos metodologías analizadas, SCRUM es un marco de trabajo que se adecua más a nuestro caso.

Tabla 2 Cuadro comparativo de metodología SCRUM y RUP

CARACTERÍSTICAS	RUP	SCRUM
ENFOQUE	Iterativo	Iterativo
CICLO	Ciclo formal se define a través de 4 fases, pero algunos flujos de trabajo pueden ser concurrentes.	Cada sprint (iteración) es un ciclo completo.
PLANIFICACIÓN	Plan de proyecto formal, asociada a múltiples iteraciones, se utiliza. El plan es impulsado fecha final y también cuenta con hitos intermedios.	No de extremo a extremo del plan del proyecto. Cada plan de la siguiente iteración se determina al final de la iteración actual (no la fecha final de tracción). Dueño del Producto (usuario de negocios clave) determina el momento en que el proyecto se lleva a cabo.
ALCANCE	Ámbito de aplicación está predefinido antes del inicio del proyecto y se documenta en el documento de Alcance. Ámbito de aplicación pueden ser revisados durante el proyecto, los requisitos se están aclarando, pero estas modificaciones están sujetas a un procedimiento estrictamente controlado.	En vez de alcance, SCRUM utiliza una Cartera de Proyectos, que se re-evaluado al final de cada iteración (sprint).
LOS ARTEFACTOS	Visión / Ámbito de aplicación del documento, el paquete formal de requisitos funcionales, documento de arquitectura del sistema, plan de	El único artefacto formal es el software operativo.

	desarrollo, plan de pruebas, scripts de prueba, etc	
TIPO DE PROYECTO / PRODUCTO	Recomendado para grandes, a largo plazo, a nivel de empresa con proyectos a medio y alta complejidad.	Recomendado para las mejoras rápidas y organizaciones que no dependen de una fecha límite.

Fuente: Elaboración Propia

Existen diversos casos de éxito de empresas que han implementado SCRUM no solo para el desarrollo de aplicaciones sino también para la estructuración de sus empresas. Entre esas empresas están SPOTIFY, Adobe Systems, Google, IBM y otras.

4.2. Implementación de la Metodología

Se definieron roles para la implementación de la presente tesis y se cumplieron con algunas actividades de la metodología elegida (SCRUM).

Roles

- **Dueño del producto (*Product Owner*):**
Raúl Ocola Gadea
Walter Ramos Mendoza
- **Equipo de desarrollo (*Development Team*):**
Raúl Ocola Gadea (Hardware + Software Arduino)
Walter Ramos Mendoza (Aplicación Móvil)

Eventos

- **Reunión de planificación (*Sprint Planning Meeting*):**
Se han realizado reuniones semanales para planear el trabajo a desarrollar durante el Sprint o evento. Se decidió qué características se iban a desarrollar y lo que era necesario para llevar a cabo el trabajo.

Para el caso del desarrollo del hardware se definieron los sensores y materiales a usar y el presupuesto para el mismo. Asimismo, se determinó quien se haría responsable del desarrollo del mismo.

Para el caso del desarrollo de la aplicación móvil se definió la el framework a usar (IONIC), el gestor de base de datos a usar (SQL Server), la forma en la que se iba a interactuar entre el hardware y software (Servicio Web) y donde debía publicarse (Servidor Apache), para el uso del mismo.

- **Scrum Diario (Daily Scrum):** Se realizaron reuniones diarias de entre 10 a 15 minutos en la cual se comentaba el avance y se definía un plan para el día. Asimismo, comentábamos las incidencias que teníamos del desarrollo de los objetivos planteados.

- **Artefactos Backlog**

No se usaron artefactos, por la limitación de tiempo que se tiene.

4.3. Presupuesto

Tabla 3 Recursos Humanos

Cargo	Cantidad	Tiempo (Meses)	Dedicación	Costo Mensual (S/.)	Total (S/.)
Desarrollador (Hardware + Software Arduino)	1	3	100%	1000.00	3000.00
Desarrollador (Aplicación Móvil)	1	3	100%	1000.00	3000.00
Total (S/.)					6000.00

Fuente: Elaboración Propia

Tabla 4 Presupuesto de Recursos de Hardware y Software

Recursos	Cantidad	Costo Unitario (S/.)	Subtotal (S/.)
Servidor Web + Base Datos			
Hosting	1	260.00	260.00
Módulo electrónico Arduino			
Sensor de medición de pH	1	70.00	70.00
Sensor de temperatura.	1	10.00	10.00
Sensor de turbidez	1	70.00	70.00
Placa Arduino Mega 2560.	1	90.00	90.00
Total (S/.)			420.00

Fuente: Elaboración Propia

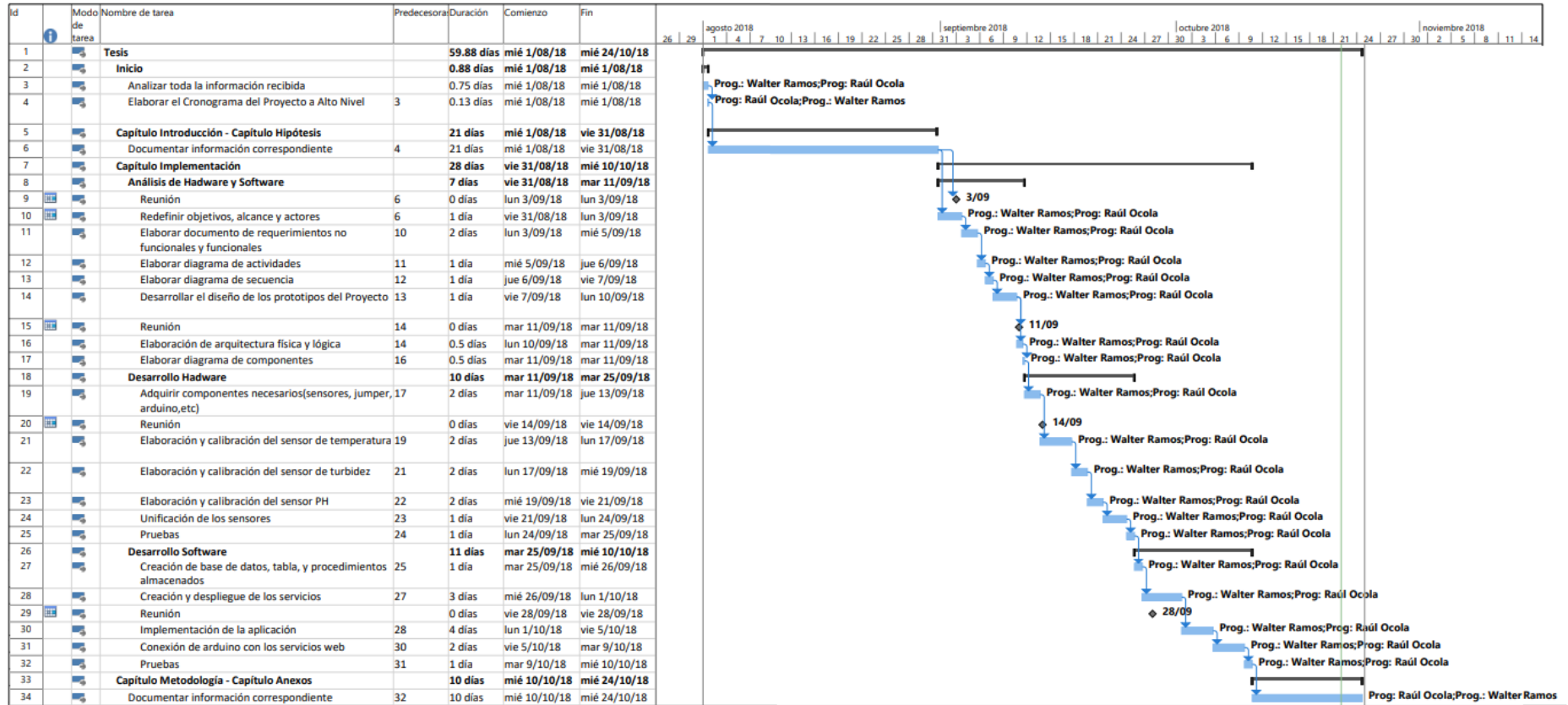
Tabla 5 Material de Escritorio

Material	Cantidad	Costo Unitario (S/.)	Subtotal (S/.)
Papel Bond (Millar)	1	12.00	12.00
Lapiceros	2	1.00	2.00
Corrector Ortográfico	2	1.50	3.00
Perforador	1	4.00	4.00
Fólder de Manila	6	1.00	6.00
Total (S/.)			27.00

Fuente: Elaboración Propia

4.4. Cronograma

Figura 12 Cronograma de actividades



Fuente: Elaboración Propia

4.5. REUNIÓN DE PLANIFICACIÓN DEL SPRINT 1 – 03/09/2018

La reunión se basa en definir y cumplir con las iteraciones establecidas en el cronograma. Además de llevar un procedimiento adecuado para la elaboración.

4.5.1. Asistentes

- Raúl Ocola Gadea (Product Owner - Development Team)
- Walter Ramos Mendoza (Product Owner - Development Team)

4.5.2. ¿Qué se determina en este Sprint? (Sprint Backlog)

En este sprint se logrará realizar el análisis y diseño de lo siguiente:

- Identificación de actores
- Caso de uso
- Requerimientos no funcionales y funcionales
- Diagrama de actividades

4.5.3. ¿Qué procedimiento se debe seguir para cumplir con estas tareas?

En el Sprint Diario, el equipo estimará el avance que se tendrá cada día según lo definido en el cronograma, así mismo no debe considerar dejar tareas pendientes para continuar con otra tarea.

4.6. Requerimientos no Funcionales y Funcionales

4.6.1. Requerimientos no Funcionales

1. Velocidad en el Tiempo de Respuesta

Tabla 6 Requerimiento no Funcional – Velocidad en el tiempo de respuesta

DESCRIPCIÓN DETALLADA
<ul style="list-style-type: none">• La aplicación móvil deberá tener un tiempo de respuesta adecuado, para cuando cualquier usuario interactúe con él.
FACTORES CRÍTICOS A CONSIDERAR
<ul style="list-style-type: none">• La aplicación móvil demora en mostrar información de acuerdo a una petición hecha por el usuario.
COMENTARIOS

Fuente: Elaboración Propia

2. Usabilidad

Tabla 7 Requerimiento no Funcional - Usabilidad

DESCRIPCIÓN DETALLADA
<ul style="list-style-type: none"> • Interfaz gráfica amigable que respete los colores y diseño del manual de marca.
FACTORES CRÍTICOS A CONSIDERAR
<ul style="list-style-type: none"> • El usuario tiene dificultades en el manejo del sistema, ya que no es intuitivo.
COMENTARIOS

Fuente: Elaboración Propia

3. Restricciones Técnicas Derivadas del Entorno Tecnológico

Tabla 8 Requerimiento no Funcional – Restricciones técnicas derivadas del entorno tecnológico

DESCRIPCIÓN DETALLADA
<ul style="list-style-type: none"> • Los servicios serán desarrollados en el lenguaje de programación JAVA, en el entorno de Eclipse. • La base de datos será desarrollada en SQL Server 2016. • La aplicación móvil será desarrollada con la herramienta IONIC, un framework basado en aplicaciones híbridas con JavaScript, Angular, Node.js.
FACTORES CRÍTICOS A CONSIDERAR
COMENTARIOS

Fuente: Elaboración Propia

4.6.2. Requerimientos funcionales

1. Registrar Parámetros de calidad

Tabla 9 Requerimiento Funcional – CU01_Registrar Parámetros de calidad

DESCRIPCIÓN BREVE	
Este caso de uso permite registrar los parámetros evaluados.	
ACTORES	
Arduino	
TRIGGER (DISPARADOR)	
Horario programado	
PRECONDICIONES	
<ul style="list-style-type: none"> Se debe tener una conexión a internet 	
FLUJO PRINCIPAL	
Paso	Acción
1	La herramienta captura los datos de los sensores.
2	La herramienta valida si cumple con la hora programada.
3	La herramienta envía la petición con los parámetros respectivos
4	El servicio valida y guarda los datos.
FLUJOS ALTERNATIVOS	
Paso	Acción
2.1	Hora no programada 2.1.1. La hora que realiza la obtención de datos, no es igual a la programada 2.1.2. La herramienta no envía la petición. 2.1.3. El caso de uso termina.
POSTCONDICIONES	
<ul style="list-style-type: none"> Se registra los valores de los parámetros en la base de datos. 	
EXTENSIONES	
REGLAS DE NEGOCIO	
REQUISITOS NO FUNCIONALES	
<ul style="list-style-type: none"> 01_Velocidad en el Tiempo de Respuesta 02_Usabilidad 03_Restricciones Técnicas Derivadas del Entorno Tecnológico 	
FACTORES CRÍTICOS A CONSIDERAR	
<ul style="list-style-type: none"> La herramienta no tiene acceso a internet. 	
COMENTARIOS	

Fuente: Elaboración Propia

2. Consultar Resultados

Tabla 10 Requerimiento Funcional – CU02_Consultar Resultados

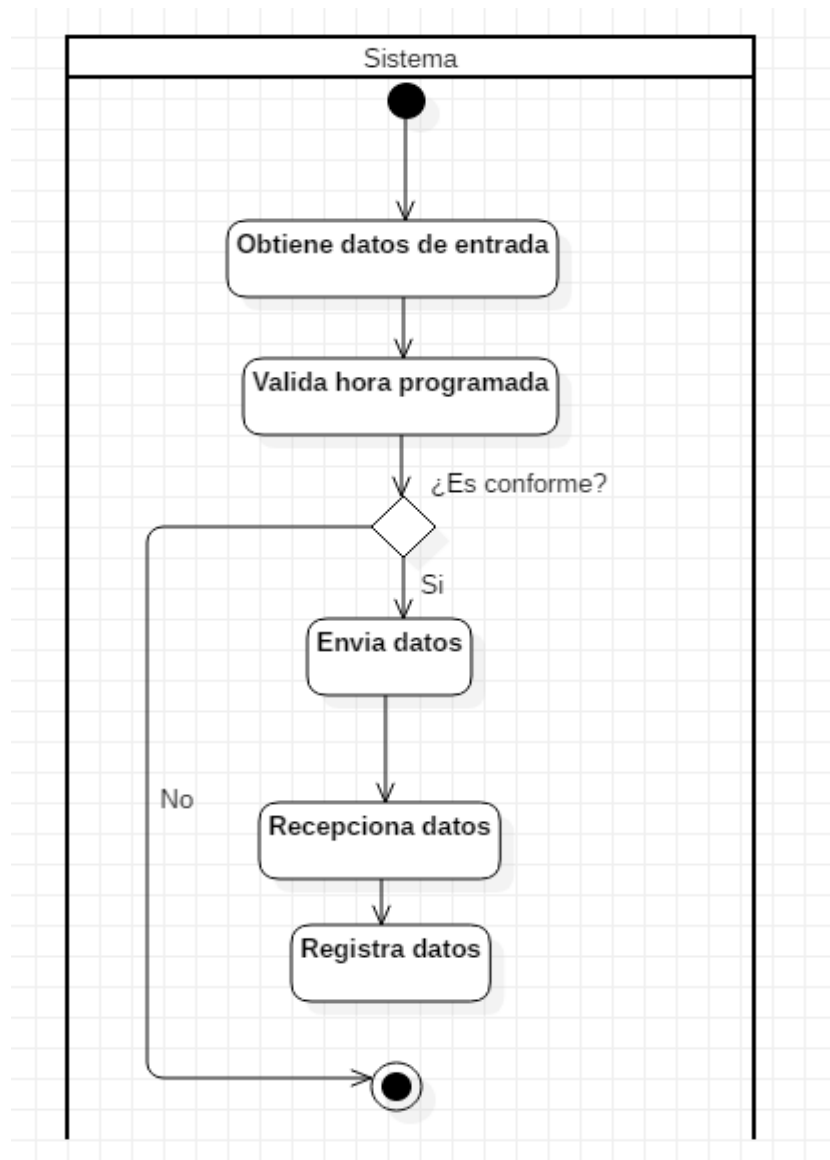
DESCRIPCIÓN BREVE	
Este caso de uso permite consultar los resultados, de la evaluación que se realizó.	
ACTORES	
Usuario	
TRIGGER (DISPARADOR)	
El usuario hace selecciona la opción “Ver Resultados”	
PRECONDICIONES	
<ul style="list-style-type: none"> • El usuario deberá acceder a la aplicación móvil. • Se debe tener una conexión a internet. 	
FLUJO PRINCIPAL	
Paso	Acción
1	El usuario selecciona la opción “Ver Resultados”
2	La aplicación móvil muestra una lista de los resultados registrados.
3	El usuario selecciona el ícono “ver”.
4	La aplicación móvil muestra una ventana emergente con el detalle de la evaluación.
FLUJOS ALTERNATIVOS	
Paso	Acción
1.1	Información 2.1.1. El usuario selecciona la opción “Información” 2.1.2. La aplicación móvil, muestra un detalle de información para el uso del software. 2.1.3. El caso de uso termina.
POSTCONDICIONES	
<ul style="list-style-type: none"> • Se visualiza los resultados. 	
EXTENSIONES	
REGLAS DE NEGOCIO	
REQUISITOS NO FUNCIONALES	
<ul style="list-style-type: none"> • 01_Velocidad en el Tiempo de Respuesta • 02_Usabilidad • 03_Restricciones Técnicas Derivadas del Entorno Tecnológico 	
FACTORES CRÍTICOS A CONSIDERAR	
<ul style="list-style-type: none"> • La herramienta no tiene acceso a internet. 	
COMENTARIOS	

Fuente: Elaboración Propia

4.7. Diagrama de Actividades

4.7.1. Registrar parámetros de calidad

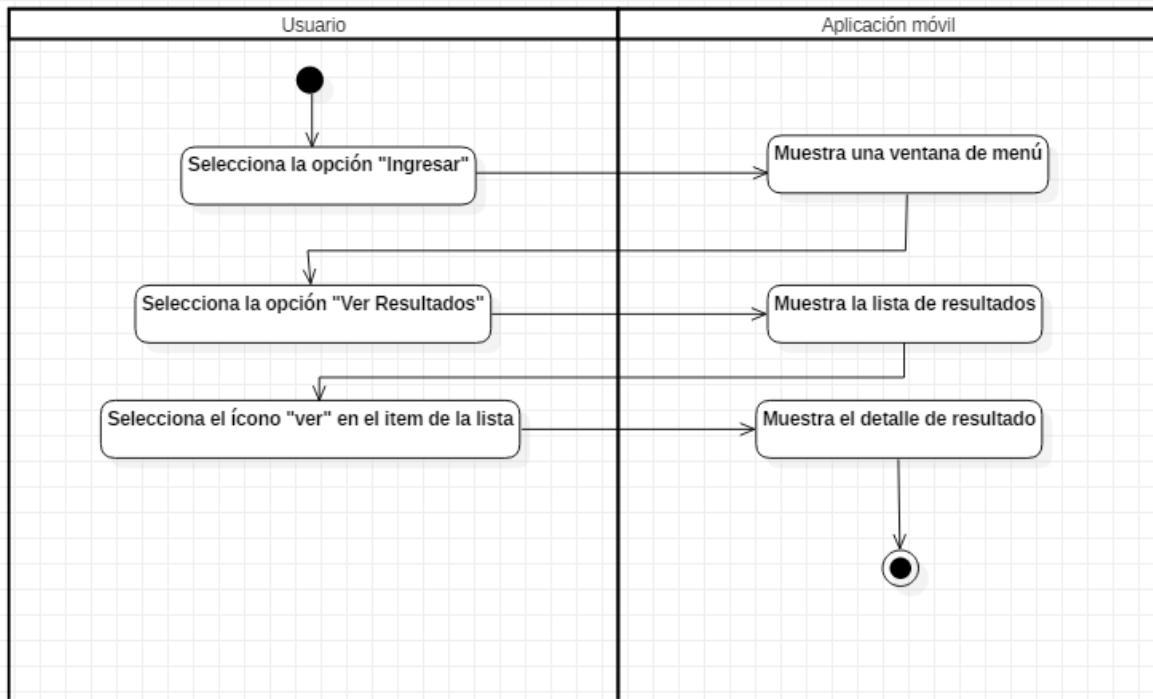
Figura 13 Diagrama de Actividades – Registrar parámetros de Calidad



Fuente: Elaboración Propia

4.7.2. Consultar Resultados

Figura 14 Diagrama de Actividades – Consultar Resultados



Fuente: Elaboración Propia

4.8. REUNIÓN DE PLANIFICACIÓN DEL SPRINT 2 – 11/09/2018

La reunión se basa en definir y cumplir con las iteraciones establecidas en el cronograma. Además de llevar un procedimiento adecuado para la elaboración.

4.8.1. Asistentes

- Raúl Ocola Gadea (Product Owner - Development Team)
- Walter Ramos Mendoza (Product Owner - Development Team)

4.8.2. ¿Qué se determina en este Sprint? (Sprint Backlog)

En este sprint se logrará realizar el análisis y diseño de lo siguiente:

- Diagrama de secuencia
- Prototipos
- Diseño de arquitectura física
- Diseño de arquitectura lógica
- Diagrama de componentes
- Tablas
- Procedimientos almacenados

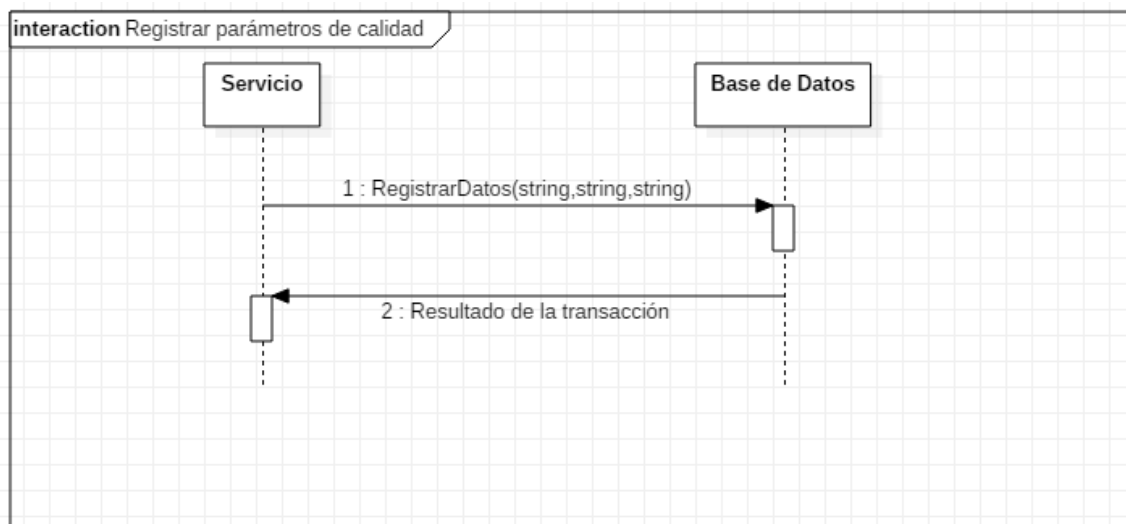
4.8.3. ¿Qué procedimiento se debe seguir para cumplir con estas tareas?

En el Sprint Diario, el equipo estimará el avance que se tendrá diariamente según lo definido en el cronograma, así mismo no debe considerar dejar tareas pendientes para continuar con otra tarea.

4.9. Diagrama de Secuencia

4.9.1. Registrar parámetros de calidad

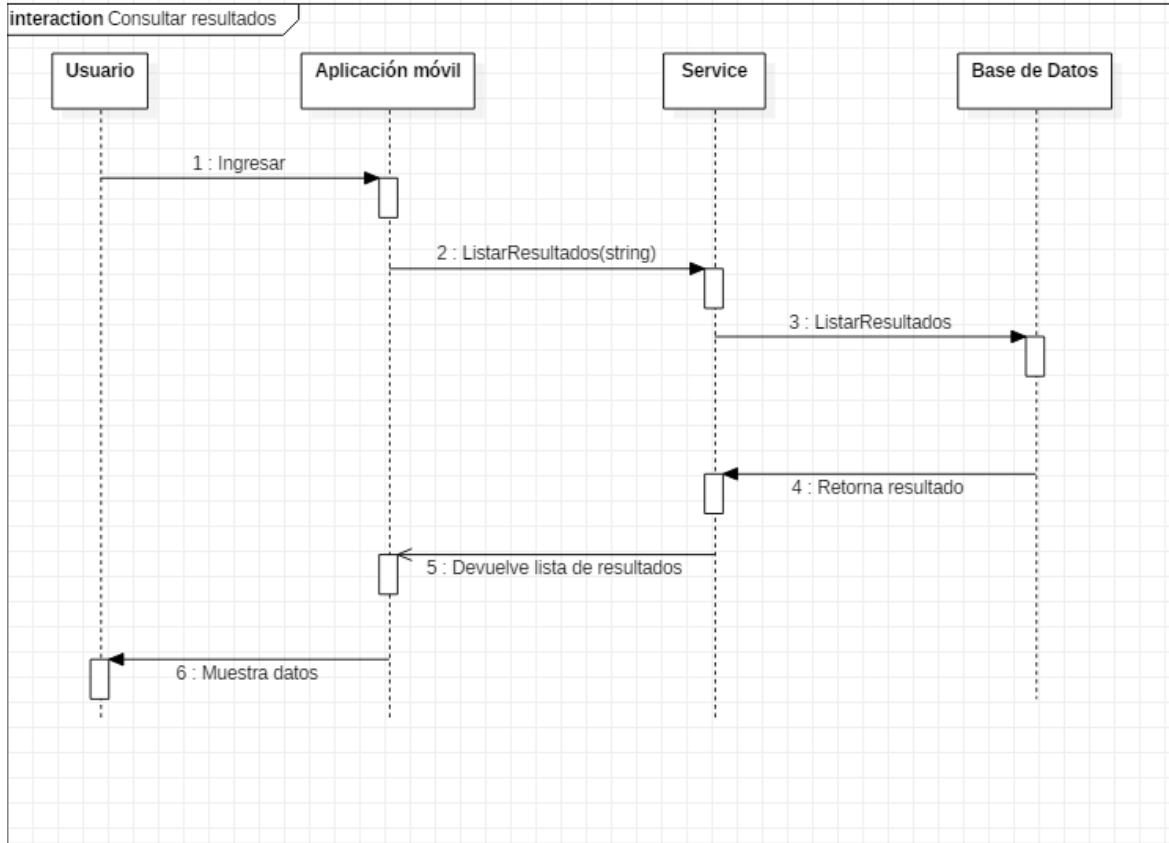
Figura 15 Diagrama de Secuencia – Registrar Parámetros de Calidad



Fuente: Elaboración Propia

4.9.2. Consultar resultados

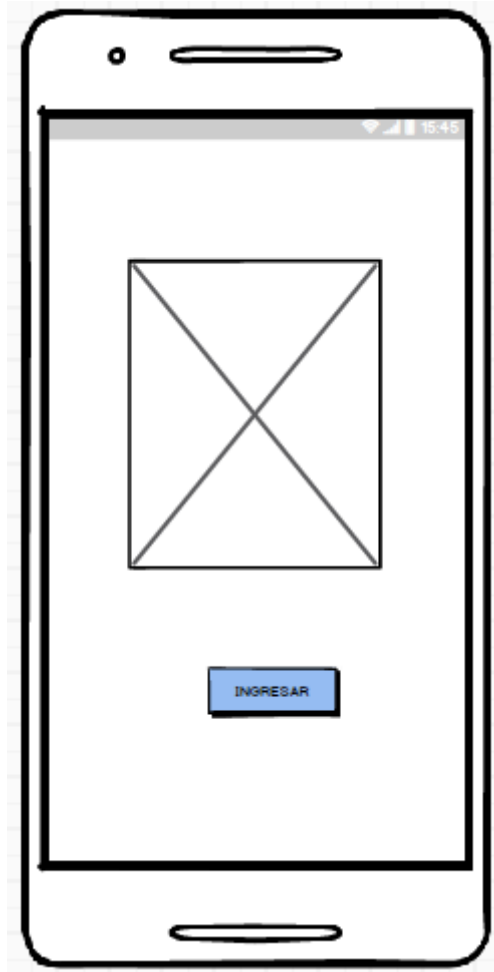
Figura 16 Diagrama de Secuencia – Consultar Resultados



Fuente: Elaboración Propia

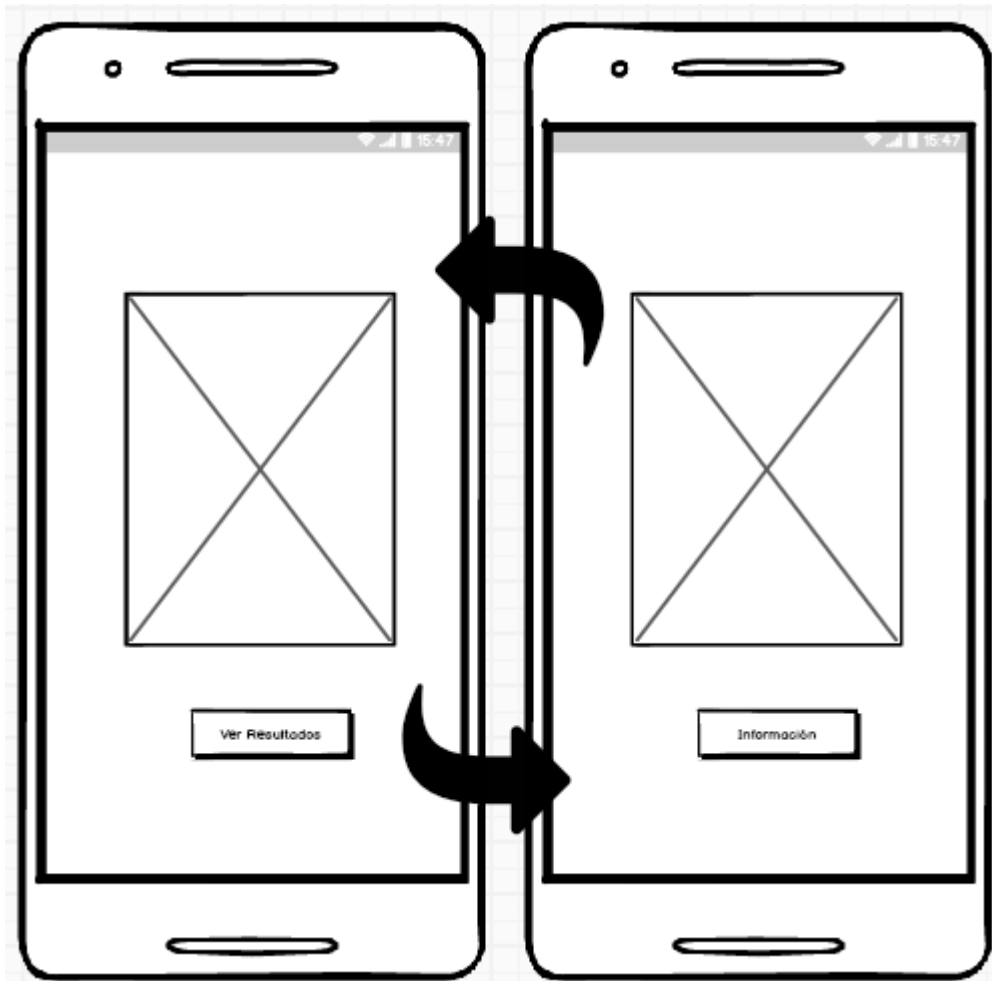
4.10. Diseño de Prototipos

Figura 17 Interfaz de Pantalla inicial de Aplicación Móvil



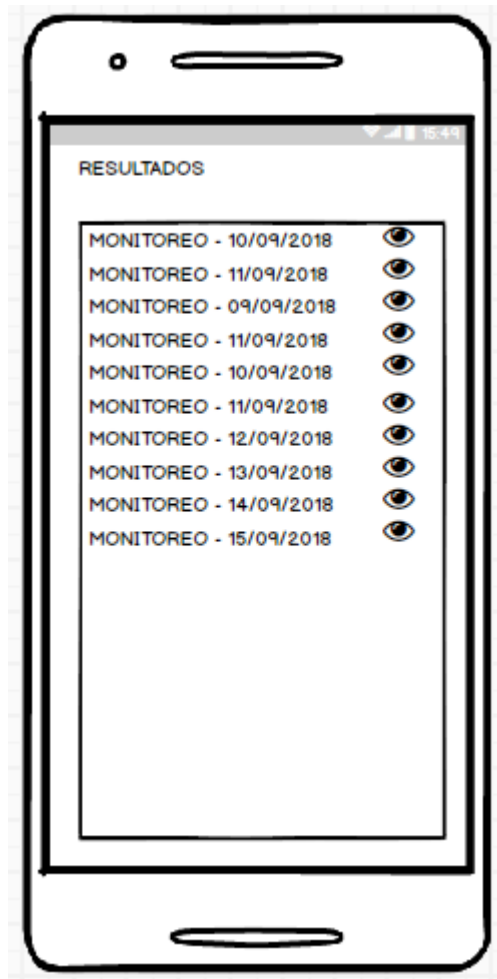
Fuente: Elaboración Propia

Figura 18 Interfaz de resultados de monitoreo e Información de la aplicación móvil



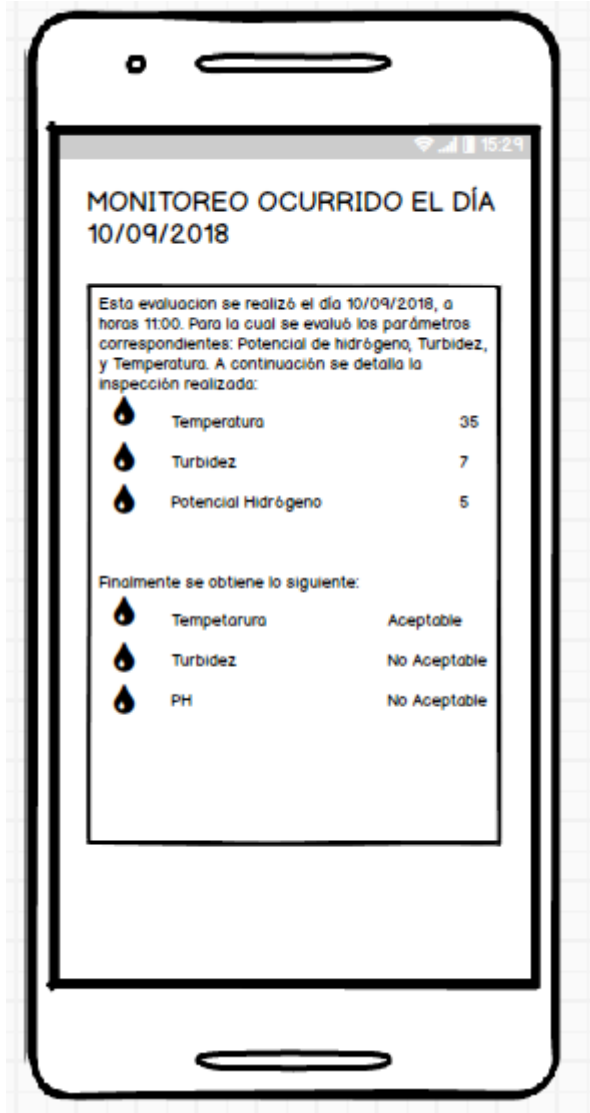
Fuente: Elaboración Propia

Figura 19 Interfaz de Lista de Resultados de Monitoreo



Fuente: Elaboración Propia

Figura 20 Interfaz de Resultado del Monitoreo del día DD/MM/AAAA

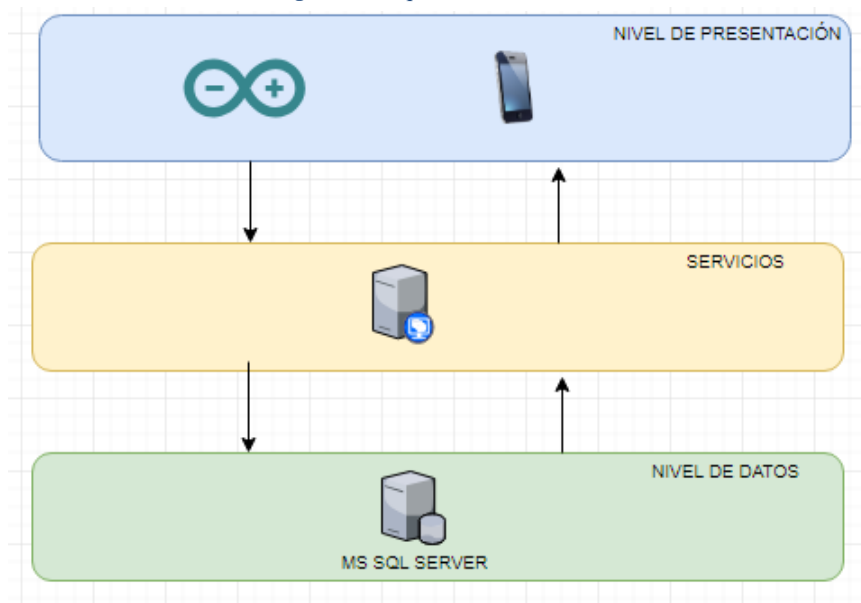


Fuente: Elaboración Propia

4.11. Diseño de Arquitectura

4.11.1. Arquitectura Física

Figura 21 Arquitectura Física



Fuente: Elaboración Propia

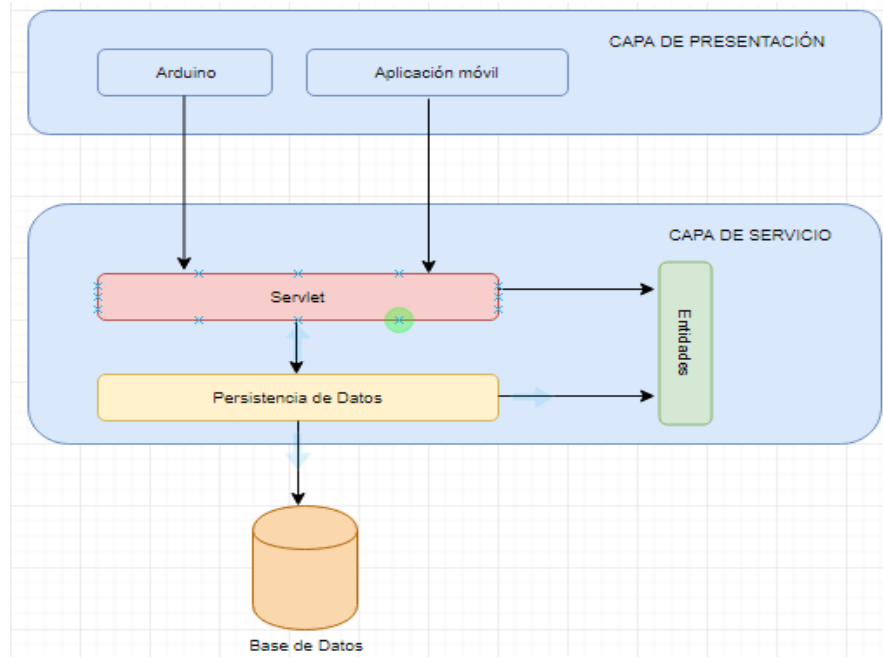
Tabla 11 Descripción Arquitectura Física

Nivel	Descripción
Presentación	Este nivel es la que el usuario ve y la conforma la aplicación móvil y Arduino. Desarrollados con el framework de Ionic y la programación de Arduino (c++).
Servicio	Este nivel permite intercambiar datos entre diversas aplicaciones. Es decir, las aplicaciones de software, pueden utilizar los servicios web para intercambiar datos.
Datos	Es un elemento lógico de la pila de tecnología que ofrece la experiencia interactiva del cliente en su sitio web. Se Desarrolla en la tecnología del Transact-SQL Server 2016.

Fuente: Elaboración Propia

4.11.2. Arquitectura Lógica

Figura 22 Arquitectura Lógica



Fuente: Elaboración Propia

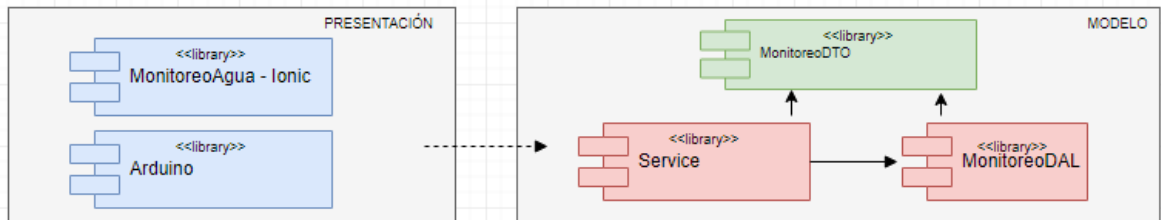
Tabla 12 Descripción de Arquitectura Lógica

Capa	Descripción
Presentación	Esta capa es la que el usuario ve y la conforma la aplicación móvil.
Servicios	<p>Esta es la capa más importante de la aplicación, aquí se definen las siguientes sub capas:</p> <ul style="list-style-type: none"> - Servlet: En esta sub capa se define las peticiones(métodos) que se hará desde una aplicación externa, así mismo de define la notación de los valores de entrada y de salida. Esta sub capa permite interactuar con la subcapa de datos. - Acceso a Datos: Esta sub capa permite la interacción entre la lógica de la aplicación y su base de datos. - Entidades: En esta sub capa se definen los DTOs (entidades) que utilizara la aplicación para el registro y consultas.
Datos	Esta capa se refiere al servidor de base de datos que empleara la aplicación, para nuestro caso se emplea el motor de base de datos MS SQL Server 2016 edición estándar.

Fuente: Elaboración Propia

4.12. Diagrama de Componentes

Figura 23 Diagrama de Componentes



Fuente: Elaboración Propia

Tabla 13 Descripción Diagrama de Componentes

Componente	Descripción
MonitoreoDAL	Este componente del tipo librería pertenece a la capa de Modelo. Aquí se encuentran las Clases que interactúan directamente con la base de datos. Depende de los componentes: MonitoreoDTO
MonitoreoDTO	Este componente del tipo librería pertenece a la capa de Modelo. Aquí el objeto transporta datos entre proceso, interactúan directamente con la librería de Service y MonitoreoDAL.
Service	Este componente permite intercambiar datos entre diversas aplicaciones. Es decir, las aplicaciones de software, pueden utilizar los servicios web para intercambiar datos.
MonitoreoAgua (Ionic)	Este componente permite crear y diseñar el contexto de la aplicación móvil, para la cual servirá para mostrar la información.
Arduino	Este componente es la que permite enviar los datos obtenidos al servicio, para que este guarde los datos. En esta capa, la herramienta Arduino, es la encargada de realizar aquella acción.

Fuente: Elaboración Propia

4.13. Base de Datos

Tabla 14 Descripción Tabla Mont_Agua_Tesis

Nombre Tabla:		MON_AGUA_TESIS						
Descripción:		Esta tabla tiene la información de los valores obtenidos mediante los sensores (Arduino) para el control de calidad de agua potable.						
Nombre Campo	Llave primaria	Tipo	V. por Defecto	Longitud	Prec	Valor Nulo	Descripción	Valores
ID	PK	int				No	Identificador único de la tabla, campo autogenerated.	
DESCRIPCION		varchar		100		Si	Breve descripción del registro	
TEMPERATURA		varchar		15		Si	Parámetro obtenido del sensor	
TURBIDEZ		varchar		15		Si	Parámetro obtenido del sensor	
PH		varchar		15		Si	Parámetro obtenido del sensor	
FECHA		datetime				Si	Fecha del registro	
HORA		time		7		Si	Hora del registro	

Fuente: Elaboración Propia

4.14. Procedimientos Almacenados

4.14.1. MONT_INSERTAR_RESULTADOS_SP

Tabla 15 Descripción Procedimiento almacenado MONT_INSERTAR_RESULTADOS_SP

SP		MONT_INSERTAR_RESULTADOS_SP			
Descripción:		Insertar resultados.			
Parámetro	Tipo	Longitud	Defecto	Descripción	Valores
TEMPERATURA	varchar	15		Valor obtenido desde el sensor.	
TURBIDEZ	varchar	15		Valor obtenido desde el sensor.	
PH	varchar	15		Valor obtenido desde el sensor.	

Fuente: Elaboración Propia

4.14.2. MONT_LISTAR_RESULTADOS_SP

Tabla 16 Descripción Procedimiento Almacenado MONT_LISTAR_RESULTADOS_SP

SP	MONT_LISTAR_RESULTADOS_SP				
Descripción:	Listar los resultados para su respectivo monitoreo.				
Parámetro	Tipo	Longitud	Defecto	Descripción	Valores
TEMPERATURA	varchar	15		Valor obtenido desde el sensor.	
TURBIDEZ	varchar	15		Valor obtenido desde el sensor.	
PH	varchar	15		Valor obtenido desde el sensor.	

Fuente: Elaboración Propia

4.15. REUNIÓN DE PLANIFICACIÓN DEL SPRINT 3 – 14/09/2018

La reunión se basa en definir y cumplir con las iteraciones establecidas en el cronograma. Además de llevar un procedimiento adecuado para la elaboración.

4.15.1. Asistentes

- Raúl Ocola Gadea (Product Owner - Development Team)
- Walter Ramos Mendoza (Product Owner - Development Team)

4.15.2. ¿Qué se determina en este Sprint? (Sprint Backlog)

En este sprint se logrará realizar lo siguiente:

- Adquirir componentes necesarios (sensores, jumper, Arduino, etc.)
- Elaboración y calibración del sensor de temperatura
- Elaboración y calibración del sensor de turbidez
- Elaboración y calibración del sensor PH
- Unificación de los sensores
- Pruebas

4.15.3. ¿Qué procedimiento se debe seguir para cumplir con estas tareas?

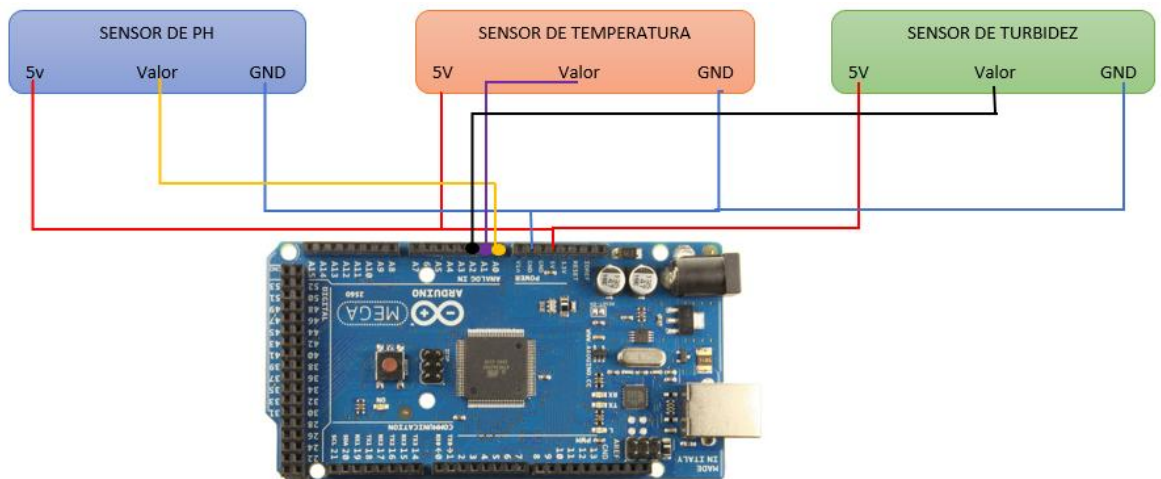
En el Sprint Diario, el equipo estimará el avance que se tendrá diariamente según lo definido en el cronograma, así mismo no debe considerar dejar tareas pendientes para continuar con otra tarea.

4.16. Implementación de Hardware

Para la implementación del Hardware se realizaron las conexiones de los módulos de los sensores DFRobot, que facilitaron la conexión del Arduino Mega 2560 a los sensores utilizados, lo cual podemos apreciar en las siguientes imágenes adjuntas:

Las conexiones correspondientes para cada sensor, se muestran en el siguiente diagrama:

Figura 24: Conexiones de sensores y Arduino Mega



Fuente: Elaboración Propia

Para la implementación de sensor de pH se conecta el sensor a su módulo DFRobot que tiene ya implementada su configuración electrónica y permite tener listo el sensor para ser programado en el IDE de Arduino, conectando solo al pin Análogo 0.

Figura 25 Implementación de Sensor de pH



Fuente: Elaboración Propia

Para la implementación del sensor de temperatura se conecta al pin Análogo 1 del Arduino y está listo el sensor para ser programado en el ID.

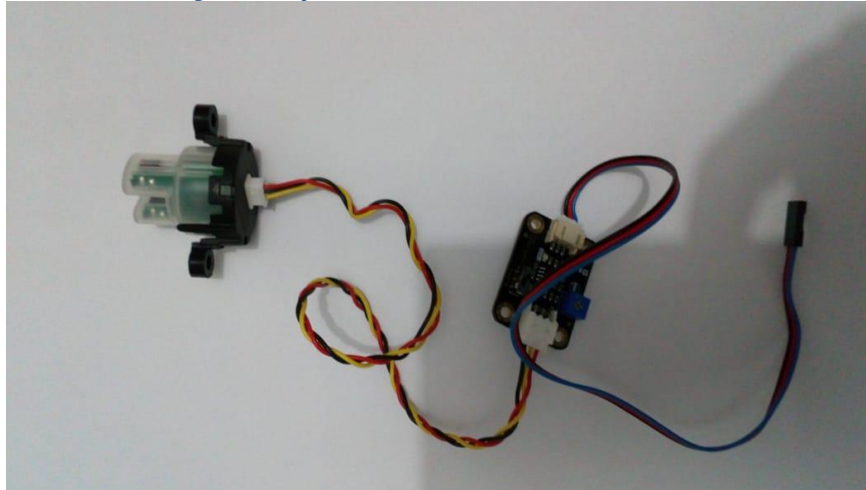
Figura 26 Sensor de Temperatura



Fuente: Elaboración Propia

Para la implementación del sensor de Turbidez se conecta a módulo DFRobot que tiene ya implementada su configuración electrónica y permite tener listo el sensor para ser programado en el IDE de Arduino, conectando solo al pin Análogo 2.

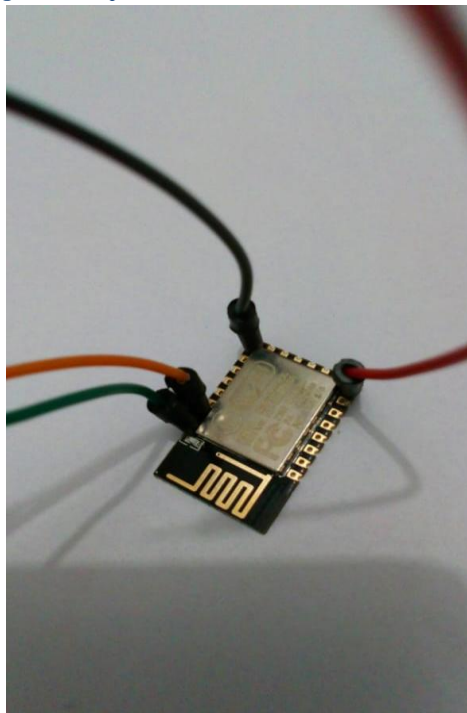
Figura 27 Implementación de Sensor Turbidez



Fuente: Elaboración Propia

Para la implementación del módulo de WIFI 8266 ESP-12, se soldaron los pines de su alimentación y para su configuración en el Arduino (Rx y Tx), quedando de la siguiente manera:

Figura 28 Implementación de Modulo WIFI 8266 ESP-12



Fuente: Elaboración Propia

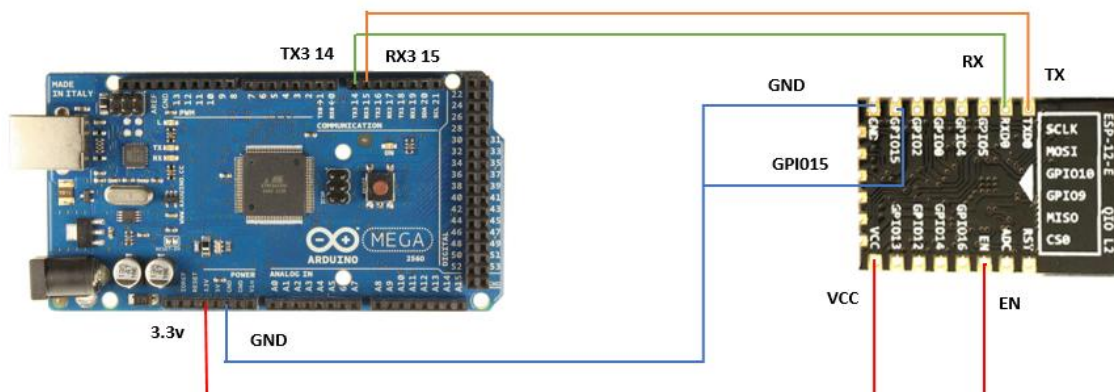
Para conectar este módulo ESP8266 12E con el Arduino, es necesario realizar las conexiones.

Tabla 17: Conexiones de ESP8266 12E con Arduino mega

ESP8266 12E		Arduino Mega 2560
VCC	→	3.3v
EN	→	3.3v
GND	→	GND
GPI015	→	GND
TX	→	RX3 15
RX	→	TX3 14

Fuente: Elaboración Propia

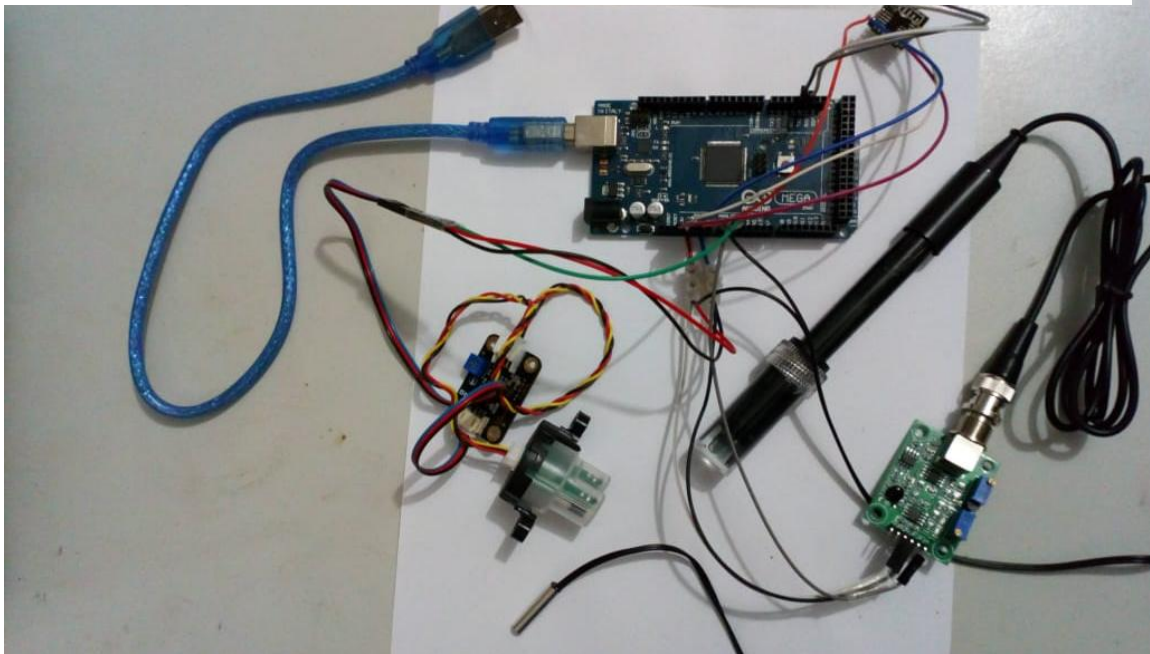
Figura 29: Conexión entre ESP8266 y Arduino Mega



Fuente: Elaboración Propia

Finalmente, una vez implementado lo anteriormente mencionado, el dispositivo queda como podemos observar a continuación:

Figura 30: Conexión completa del dispositivo



Fuente: Elaboración Propia

El agua que alimenta el pozo donde se va implementar el prototipo, llega antes a una sección donde es idóneo poner el equipo antes mencionado, ya que en el pozo en sí se realiza el proceso de cloración que sirve para asegurar que el agua este apta para el consumo humano.

La sección y el pozo antes mencionados se encuentran ubicados en el centro poblado Bello Horizonte, del distrito de Laredo, los cuales podemos apreciar en las siguientes imágenes.

Figura 31 Sección donde se implementara el dispositivo



Fuente: Elaboración Propia

Figura 32 Pozo de agua



Fuente: Elaboración Propia

Figura 33 Sección donde se implementará el dispositivo y pozo de agua



Fuente: Elaboración Propia

4.17. REUNIÓN DE PLANIFICACIÓN DEL SPRINT 4 – 28/09/2018

La reunión se basa en definir y cumplir con las iteraciones establecidas en el cronograma. Además de llevar un procedimiento adecuado para la elaboración.

4.17.1. Asistentes

- Raúl Ocola Gadea (Product Owner - Development Team)
- Walter Ramos Mendoza (Product Owner - Development Team)

4.17.2. ¿Qué se determina en este Sprint? (Sprint Backlog)

En este sprint se logrará realizar lo siguiente:

- Creación de base de datos, tabla, y procedimientos almacenados.
- Creación y despliegue de los servicios.
- Implementación de la aplicación.
- Conexión de Arduino con los servicios web.
- Unificación de los sensores
- Pruebas

¿Qué procedimiento se debe seguir para cumplir con estas tareas?

En el Sprint Diario, el equipo estimará el avance que se tendrá diariamente según lo definido en el cronograma, así mismo no debe considerar dejar tareas pendientes para continuar con otra tarea.

4.18. Implementación de Software

Para la implementación del Software, tenemos 2 implementaciones:

- ❖ Implementación de Software para los sensores en la placa Arduino Mega 2560:

Código implementado para medir la temperatura del agua:

Figura 34 Implementación Código para medir Temperatura

```
//////////////////////////////////TEMPERATURA//////////////////////////////////  
int val;//Crea una variable entera  
double temp;//Variable de temperatura = temp  
val=analogRead(NTCPin);//Lee el valor del pin analogo 0 y lo mantiene como val  
temp=Thermister(val);//Realiza la conversión del valor analogo a grados Celsius
```

Fuente: Elaboración Propia

Código implementado para medir el pH del agua:

Figura 35 Implementación Código para medir pH

```

////////////////////////////////////PH////////////////////////////////////
for(int i=0;i<10;i++) //Obtener 10 valores del muestreo del sensor para suavizar el valor
{
    buf[i]=analogRead(SensorPH);
    delay(10);
}
for(int i=0;i<9;i++) //ordenar el análogo del más pequeño al más grande
{
    for(int j=i+1;j<10;j++)
    {
        if(buf[i]>buf[j])
        {
            temp0=buf[i];
            buf[i]=buf[j];
            buf[j]=temp0;
        }
    }
}
avgValuePH=0;
for(int i=2;i<8;i++) // acumula las medidas centrales (las 6)
    avgValuePH+=buf[i];
float phValue=(float)avgValuePH*5.0/1024/6; //Convertir el valor analógico a V (valor promedio de 6 muestras centrales)
phValue=-4.1285*phValue+19.935; //Convertir los voltios a valores de pH
    
```

Fuente: Elaboración Propia

Código implementado para medir la Turbidez del agua:

Figura 36 Implementación del código para medir la Turbidez

```

////////////////////////////////////TURB////////////////////////////////////
for(int k=0;k<10;k++) //Obtener 10 valores del muestreo del sensor para suavizar el valor
{
    buf1[k]=analogRead(SensorTURB);
    delay(10);
}
for(int k=0;k<9;k++) //ordenar el análogo del más pequeño al más grande
{
    for(int l=k+1;l<10;l++)
    {
        if(buf1[k]>buf1[l])
        {
            temp1=buf1[k];
            buf1[k]=buf1[l];
            buf1[l]=temp1;
        }
    }
}
avgValueTURB=0;
for(int k=2;k<8;k++) // acumula las medidas centrales (las 6)
    avgValueTURB+=buf1[k];
float NTUvalue=(float)avgValueTURB*5.0/1024/6*1000; //Convertir el valor analógico a mV (valor promedio de 6 muestras centrales)
NTUvalue=(4189-NTUvalue)*1.58; //Se hizo la calibración a 23 °C. Convertir los V a valores de NTU
    
```

Fuente: Elaboración Propia

Código implementado para la configuración y envío de parámetros obtenidos por los sensores, mencionados anteriormente:

Figura 37 Configuración de módulo de wifi para el envío de información

```
//COMANDOS AT PARA EL ENVIO DE INFORMACION
sendData("AT+CWMODE=3\r\n",1000,DEBUG); // configuración punto de acceso
sendData("AT+CIPMUX=1\r\n",1000,DEBUG); // multiples conexiones
sendData("AT+CIPSERVER=1,80\r\n",1000,DEBUG); // encender servidor en puerto 80
delay(10000);
sendData("AT+CWJAP=\"Walter R\", \"#1234WRm\"\r\n",1000,DEBUG); //red wifi, usuario y clave
delay(10000);
sendData("AT+CIFSR\r\n",1000,DEBUG); // se obtienen direcciones IP
delay(10000);
}
```

Fuente: Elaboración Propia

Figura 38 Código para envío de información obtenida por sensores

```
//ENVIA INFORMACIÓN AL SERVICIO
int connectionId = Serial3.read()-48; // Restar 48 porque la función read () devuelve
String cmd = "AT+CIPSTART=\"TCP\", \"\"; // Setup TCP connection
cmd += IP;
cmd += "\",9080\r\n";
sendData(cmd,1000,true);
delay(2000);
String url = "POST /RestJR/services/ProyTesis/InsertaDatos?cond=1000&ph="+String(phValue)+"&temp="+String(temp)+"&turb="+String(NTUval);
sendData("AT+CIPSEND="+ String(url.length( )) + "\r\n", 500, true);
sendData(url, 1000, true); //
//sendData("AT+CIPCLOSE="+ String(connectionId) + "\r\n", 1000, true);
//-----
```

Fuente: Elaboración Propia

- ❖ Implementación de los servicios para luego ser consumidos por la aplicación móvil y el Arduino:

Código implementado para el **Servicio** a consumir por la Aplicación Móvil:

- **Entidades**

Figura 39 Clase Entidad

```
1 package com.javarevolutions.ws.rest.vo;
2
3 import java.util.Date;
4
5 public class MonitoreoDTO {
6     private int Id;
7     private String Temperatura;
8     private String Ph;
9     private String Turbidez;
10    private Date Fecha;
11    private Date Hora;
12    private String Descripcion;
13    private String ResultTemp;
14    private String ResultTurb;
15    private String ResultPh;
16
17    public int getId() {
18        return Id;
19    }
20    public void setId(int id) {
21        Id = id;
22    }
23    public String getTemperatura() {
24        return Temperatura;
25    }
26    public void setTemperatura(String temperatura) {
27        Temperatura = temperatura;
28    }
29    public String getPh() {
30        return Ph;
31    }
32    public void setPh(String ph) {
33        Ph = ph;
34    }
35    public String getTurbidez() {
36        return Turbidez;
37    }
}
```

Fuente: Elaboración Propia

- **Datos**

Figura 40 Método para Listar Resultados de Lectura de los sensores

```
public boolean InsertarResultado(String ph, String temp, String turb ) throws Exception{
    Connection cn = Conexion.Instancia().Conectar();
    try {
        CallableStatement cst = cn.prepareCall(
            "{call MONT_INSERTAR_RESULTADOS_SP(?,?,?)}");
        cst.setString(1, temp);
        cst.setString(2, turb);
        cst.setString(3, ph);
        cst.executeUpdate();
        return true;
    } catch (Exception e) {
        throw e;
    }finally{cn.close();}
}
```

Fuente: Elaboración Propia

Figura 41 Método Insertar Resultados de lectura de sensores

```
public List<MonitoreoDTO> ListarResultados() throws Exception{
    Connection cn = null;
    List<MonitoreoDTO> lista = new ArrayList<MonitoreoDTO>();
    try {
        cn = Conexion.Instancia().Conectar();
        CallableStatement cst = cn.prepareCall("{call MONT_LISTAR_RESULTADOS_SP()}");
        ResultSet rs = cst.executeQuery();
        while (rs.next()) {
            MonitoreoDTO u = new MonitoreoDTO();
            u.setId(rs.getInt("ID"));
            u.setDescripcion(rs.getString("DESCRIPCION"));
            u.setPh(rs.getString("PH"));
            u.setTemperatura(rs.getString("TEMPERATURA"));
            u.setTurbidez(rs.getString("TURBIDEZ"));
            u.setFecha(rs.getDate("FECHA"));
            u.setHora(rs.getTime("HORA"));
            u.setResultPh(rs.getString("RESTTPH"));
            u.setResultTemp(rs.getString("RESTTEMP"));
            u.setResultTurb(rs.getString("RESTTURB"));
            lista.add(u);
        }
    } catch (Exception e) {
        throw e;
    } finally {
        cn.close();
    }
    return lista;
}
```

Fuente: Elaboración Propia

- Servicio

Figura 42 Método Listar

```
@GET
@Path("/ListaResultados")
@Consumes({MediaType.APPLICATION_JSON})
@Produces({MediaType.APPLICATION_JSON})
public VOResult ListaResultados(String wr){
    List<MonitoreoDTO> lista = new ArrayList();
    VOResult results = new VOResult();
    try {
        lista = MonitoreoDAL.Instancia().ListarResultados();
        results.setResult(lista);
    } catch (Exception e) {
        e.printStackTrace();
    }
    return results;
}
```

Fuente: Elaboración Propia

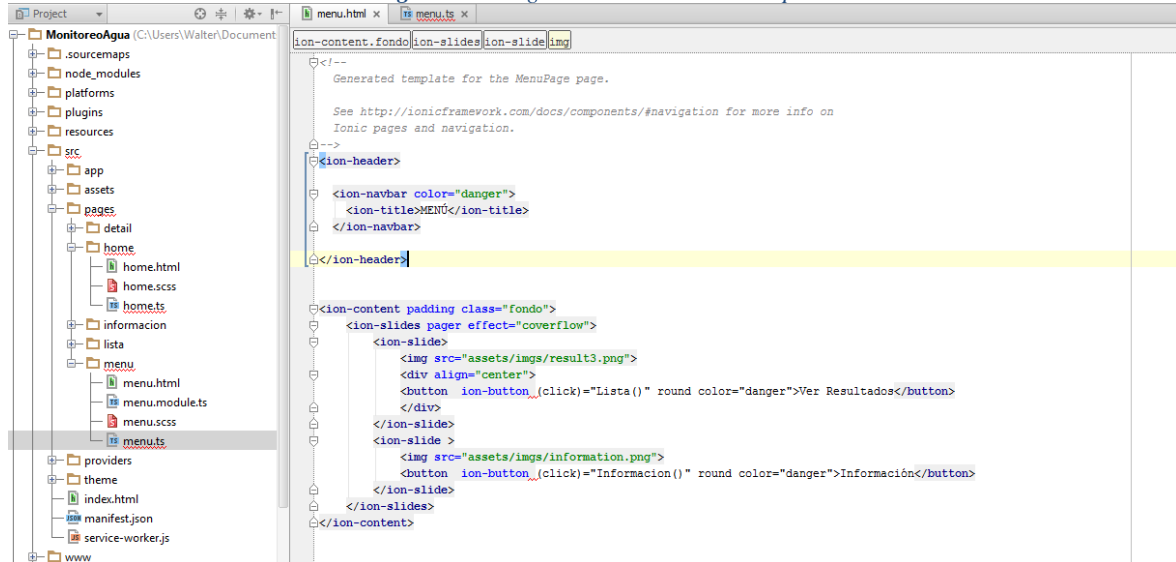
Figura 43 Método para insertar datos

```
@POST
@Path("/InsertaDatos")
@Produces({MediaType.APPLICATION_JSON})
public boolean GrabarDatos(@QueryParam("ph") String ph,@QueryParam("temp") String temp,@QueryParam("turb") String turb){
    boolean resultado = false;
    try {
        resultado = MonitoreoDAL.Instancia().InsertarResultado(ph, temp, turb);
    } catch (Exception e) {
        e.printStackTrace();
    }
    return resultado;
}
```

Fuente: Elaboración Propia

Código para la implementación de la aplicación móvil hecha con Ionic Framework:

Figura 44 Código HTML del Menú Principal



```

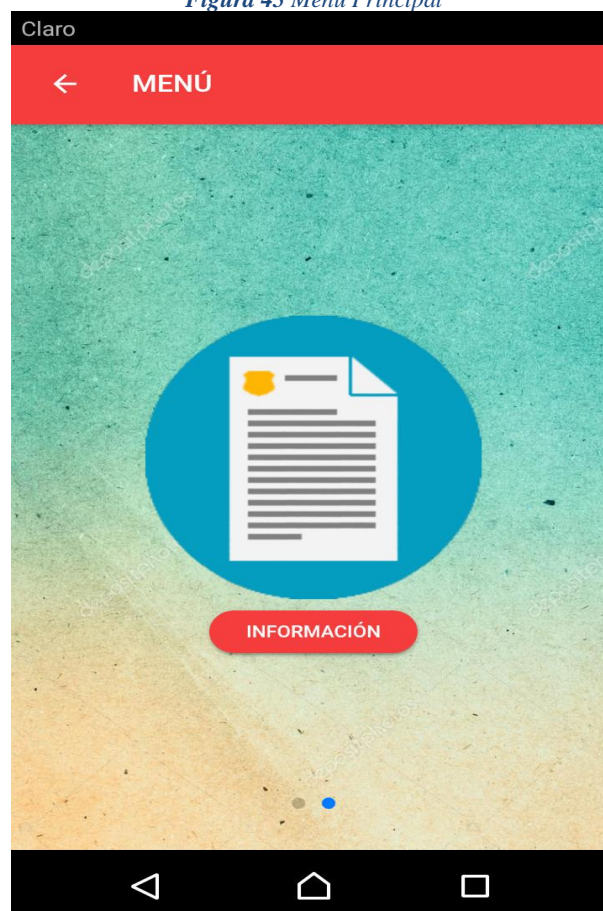
<!--
  Generated template for the MenuPage page.
  See http://ionicframework.com/docs/components/#navigation for more info on
  Ionic pages and navigation.
-->
<ion-header>
  <ion-navbar color="danger">
    <ion-title>MENÚ</ion-title>
  </ion-navbar>
</ion-header>

<ion-content padding class="fondo">
  <ion-slides pager effect="coverflow">
    <ion-slide>
      
      <div align="center">
        <button ion-button,(click)="Lista()" round color="danger">Ver Resultados</button>
      </div>
    </ion-slide>
    <ion-slide >
      
      <button ion-button,(click)="Informacion()" round color="danger">Información</button>
    </ion-slide>
  </ion-slides>
</ion-content>

```

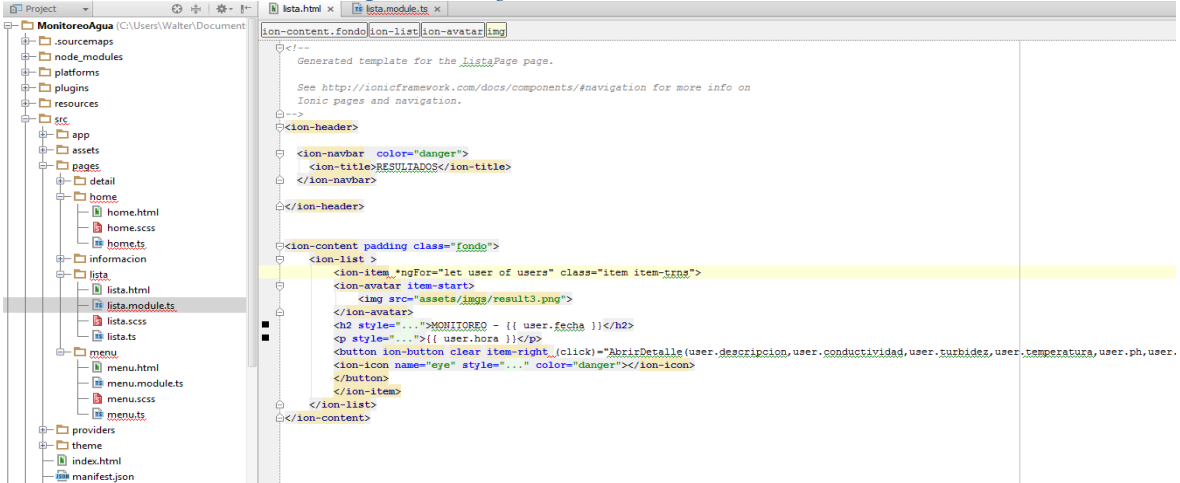
Fuente: Elaboración Propia

Figura 45 Menú Principal



Fuente: Elaboración Propia

Figura 46 Código HTML de la Lista de Resultados



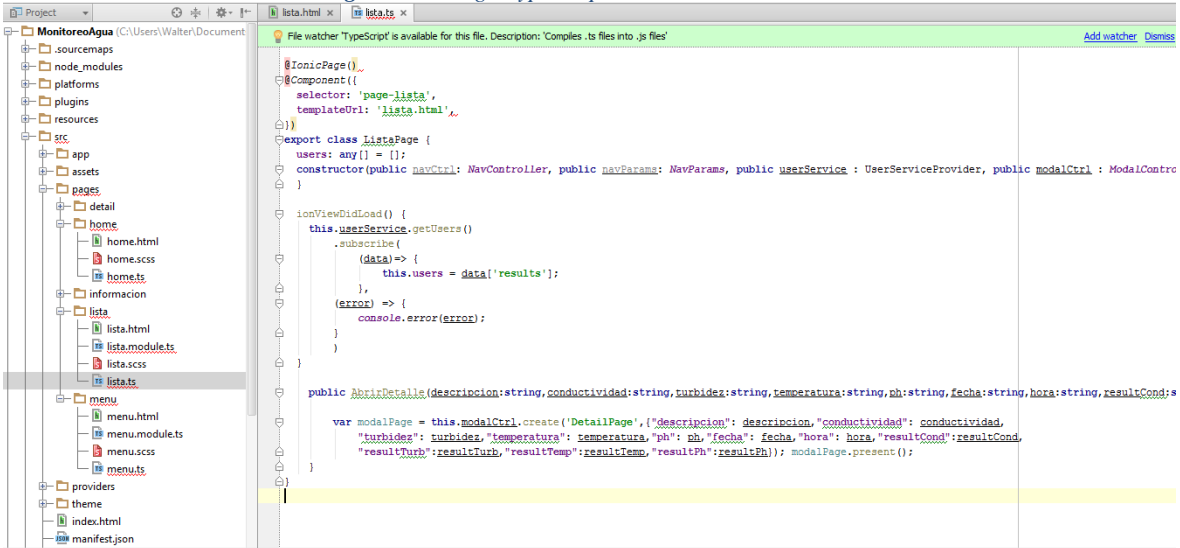
```

<!--
Generated template for the ListaPage page.
See http://ionicframework.com/docs/components/#navigation for more info on
Ionic pages and navigation.
-->
<ion-header>
  <ion-nav-bar color="danger">
    <ion-title>RESULTADOS</ion-title>
  </ion-nav-bar>
</ion-header>
<ion-content padding class="fondo">
  <ion-list >
    <ion-item *ngFor="let user of users" class="item item-trns">
      <ion-avatar item-start>
        
      </ion-avatar>
      <h2 style="...">MONITOREO - {{ user.fecha }}</h2>
      <p style="...">{{ user.hora }}</p>
      <button ion-button clear item-right (click)="AbrirDetalle(user.descripcion,user.conductividad,user.turbidez,user.temperatura,user.ph,user.fecha)"><ion-icon name="eye" style="..." color="danger"></ion-icon>
    </ion-item>
  </ion-list>
</ion-content>

```

Fuente: Elaboración Propia

Figura 47 Código TypeScript de la Lista de Resultados



```

@IonicPage()
@Component({
  selector: 'page-lista',
  templateUrl: 'lista.html'
})
export class ListaPage {
  users: any[] = [];
  constructor(public navCtrl: NavController, public navParams: NavParams, public userService: UserServiceProvider, public modalCtrl: ModalController) {}

  ionViewDidLoad() {
    this.userService.getUsers()
      .subscribe(
        (data) => {
          this.users = data['results'];
        },
        (error) => {
          console.error(error);
        }
      );
  }

  public AbrirDetalle(descripcion:string, conductividad:string, turbidez:string, temperatura:string, ph:string, fecha:string, hora:string, resultCond:string) {
    var modalPage = this.modalCtrl.create('DetallePage', {
      "descripcion": descripcion,
      "conductividad": conductividad,
      "turbidez": turbidez,
      "temperatura": temperatura,
      "ph": ph,
      "fecha": fecha,
      "hora": hora,
      "resultCond": resultCond,
      "resultTurb": resultTurb,
      "resultTemp": resultTemp,
      "resultPh": resultPh
    });
    modalPage.present();
  }
}

```

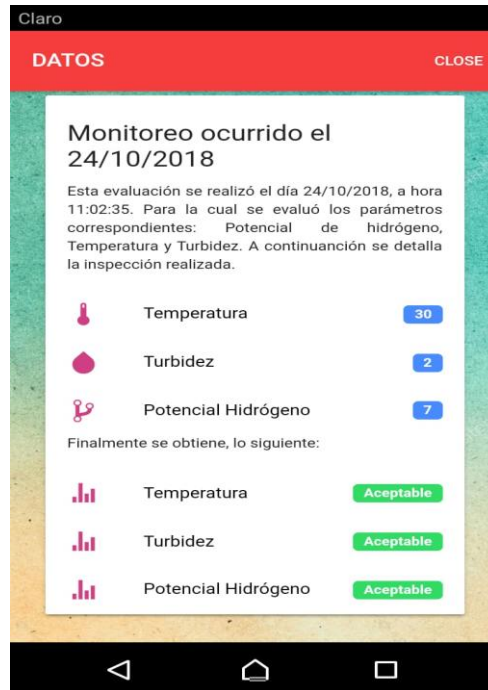
Fuente: Elaboración Propia

Figura 48 Lista de Resultados de los parámetros de calidad del agua



Fuente: Elaboración Propia

Figura 49 Datos de Monitoreo de parámetros de calidad del agua



Fuente: Elaboración Propia

Figura 50 Información de la Finalidad de la Aplicación



Fuente: Elaboración Propia

CAPÍTULO 5. METODOLOGÍA

5.1. Diseño de investigación

5.1.1. Pre Experimental

Símbolo	Significado
G	Grupo de sujetos (G1: Grupo 1)
O ₁	Medición de los sujetos de un grupo (Prueba, cuestionario, observación, etc.).
X	Tratamiento, estímulo o condición experimental
O ₂	Medición de los sujetos de un grupo (Prueba, cuestionario, observación, etc.).

5.2. Unidad de estudio

Pozos de agua potable del Distrito de Laredo.

5.3. Población

33 lecturas correspondientes al Análisis de parámetros Físicos y Químicos de los 6 Pozos de agua potable del Distrito de Laredo, Trujillo.

5.4. Muestra (muestreo o selección)

Debido a que conocemos el tamaño de la población, se procede a utilizar la fórmula expuesta a continuación para el cálculo respectivo del tamaño de la muestra.

- ✓ Z = Nivel de confianza (95%) = 1.96
- ✓ P = Probabilidad de éxito = 0.5
- ✓ Q = Probabilidad de fracaso = 0.5
- ✓ E = % del estimador o error máximo admisible = 0.05
- ✓ N = Tamaño de la población
- ✓ n = Tamaño de la muestra.

$$n = \frac{Z^2 * N * (P * Q)}{E^2 * (N - 1) + Z^2 * (P * Q)}$$

$$n = \frac{1.96^2 * 33 * (0.5 * 0.5)}{0.05^2 * (33 - 1) + 1.96^2 * (0.5 * 0.5)}$$

$$n = \frac{31.6932}{1.0404}$$

$$n = 30.4$$

n = Total de Lecturas a evaluar = 30

5.5. Técnicas, instrumentos y procedimientos de recolección de datos

Tabla 18 Técnicas, instrumentos y procedimientos.

Variable	Indicador	Técnica	Instrumento
Monitoreo de la calidad del agua potable	Nivel de Eficacia del Resultado	Observación	Lista de Observación
Aplicación móvil complementada con la tecnología de sensores Multiparamétricos con Arduino	Compleitud funcional	Observación	Lista de Observación
	Corrección funcional	Observación	Lista de Observación
	Comportamiento temporal	Observación	Lista de Observación

Fuente: Elaboración Propia

5.6. Métodos, instrumentos y procedimientos de análisis de datos

Tabla 19: Métodos, instrumentos y procedimientos

Variable	Indicador	Procedimiento	Formula
Monitoreo la calidad del agua potable	Nivel de Eficacia del Resultado	Análisis Cualitativo	$NER = RC/TM$ Donde: NCA = Nivel de Calidad del Agua RC = Nro. Resultados correctos TM = Nro. Total de muestras
Aplicación móvil complementada con la tecnología de sensores Multiparamétricos con Arduino	Compleitud funcional	Análisis Cuantitativo	$CF = FD/TFI$ Donde: CF = Compleitud funcional FD = Nro. de funciones que detectan al probar TFI = Nro. de funciones implementadas
	Corrección funcional	Análisis Cuantitativo	$UR = CR/TRD$ Donde: UR = Utilización de recursos (MB) CR = Cantidad de recursos utilizados TRD = Total de recursos disponibles
	Comportamiento temporal	Análisis Cuantitativo	Tiempo de respuesta (s.)

Fuente: Elaboración Propia

CAPÍTULO 6. RESULTADOS

6.1. Planteamiento de Hipótesis

Dada la hipótesis planteada: “La implementación de una aplicación móvil, complementada con la tecnología de sensores Multiparamétricos con Arduino permite monitorear la calidad del agua potable”.

- **Variable Dependiente:** Monitoreo de la calidad del agua potable
- **Variable Independiente:** Aplicación móvil complementada con la tecnología de sensores Multiparamétricos con Arduino

6.2. Contratación de la Hipótesis

La contratación de Hipótesis se ha realizado mediante los métodos Pre -Test y Post - Test, con los cuales se podrá rechazar o aceptar la hipótesis. Así como también se identificaron indicadores cuantitativos y cualitativos, los cuales se detallarán en la siguiente tabla:

Tabla 20 Tabla de Indicadores

N°	Indicador	Tipo
1	Nivel de Completitud Funcional	Cuantitativo
2	Nivel de Corrección Funcional	Cuantitativo
3	Nivel de Comportamiento Temporal	Cuantitativo
4	Nivel de Eficacia del Resultado	Cualitativo

Fuente: Elaboración Propia

6.3. Análisis de los Indicadores Cuantitativos

Para medir los indicadores cuantitativos en la siguiente tabla delimitaremos los valores según el nivel de acierto de los indicadores de completitud y corrección funcional.

Tabla 21 Escala de Medición Indicadores Cuantitativos

Nivel de Acierto	Valor
Cumple	1
No Cumple	0

Fuente: Elaboración Propia

6.3.1. Nivel de completitud funcional

Definición de Variables

NCFA: Nivel de completitud funcional antes de la implementación de la aplicación móvil propuesta.

NCFD: Nivel de completitud funcional después de la implementación de la aplicación móvil propuesta.

Hipótesis Estadística

Ho: NCFA – NCFD = 0

Ha: NCFA – NCFD > 0

Nivel de Significancia

El nivel de significancia (α) escogido para la prueba de la hipótesis es del 5%. Por lo tanto, el nivel de confianza ($1 - \alpha = 0.95$) será del 95%.

Estadística de Contraste

Puesto que la muestra son las lecturas del pozo seleccionado como muestra, $n=30$, usaremos la distribución normal (**ver Anexo A.**) para lo cual se tiene:

Fórmula N° 1: Promedio

$$\bar{X} = \frac{\sum_{i=1}^n X_i}{n} \dots\dots\dots(1)$$

Fórmula N° 1: Varianza de Z

$$\sigma^2 = \frac{\sum_{i=1}^n (X_i - \bar{X})^2}{n} \dots\dots\dots(2)$$

Fórmula N° 2: Prueba Z

$$Z_c = \frac{\overline{\text{NCFA}} - \overline{\text{NCFD}}}{\sqrt{\left(\frac{\sigma_A^2}{n_A} + \frac{\sigma_P^2}{n_P}\right)}} \dots\dots\dots(3)$$

Cálculos

Para calcular el nivel de completitud funcional para una muestra de 30 lecturas realizadas para los parámetros evaluados es necesario calcular los promedios del nivel de completitud funcional antes y después de la implementación de la aplicación móvil propuesta.

Nivel de completitud funcional respecto a la medición de parámetros de calidad del agua sin la aplicación móvil.

Tabla 22 Nivel de completitud funcional sin la aplicación móvil.

N° de Aciertos para Indicador de Completitud Funcional			
N°	Rangos		Acierto
	pH 6,5 - 8,5	Turbidez 0 - 5	
1	6,9	0,09	1
2	6,8	0,09	1
3	6,8	0,09	1
4	6,8	0,09	1
5	6,9	0,09	1
6	7	0,9	1
7	6,5	0,9	1
8	6,5	3	1
9	6,9	0,24	1
10	6,7	2	1
11	6,8	0,09	1
12	6,4	1,27	0
13	6,4	0,9	0
14	6,9	0,9	1
15	6,7	0,9	1
16	6,7	0,09	1
17	6,5	0,09	1
18	6,3	0,09	0
19	6,8	0,57	1
20	6,8	4,09	1
21	6,5	1,04	1
22	6,9	1,88	1
23	6,8	1,99	1
24	6,9	0,2	1

25	6,6	0,43	1
26	8	0,27	1
27	8,3	0,22	1
28	7,4	0,26	1
29	7,2	0,3	1
30	6,9	0,2	1
TOTAL			27

Fuente: Elaboración Propia

$$n_a = n_p = 30$$

Promedio

Reemplazando los valores en la fórmula $n=30$:

$$\overline{NCF\bar{A}} = \frac{27}{30}$$

$$\overline{NCF\bar{A}} = 0.9$$

Varianza:

Reemplazando los valores en la fórmula

$$\sigma_A^2 = 0.09$$

Nivel de completitud funcional respecto a la medición de parámetros de calidad del agua con la aplicación móvil

Tabla 23 Nivel de completitud Funcional con la Aplicación Móvil

N° de Aciertos para Indicador de Completitud Funcional				
N°	Rangos			Acierto
	pH 6,5 - 8,5	Turbidez 0 - 5	Temperatura 15 - 35	
1	5,9	0,4	27,0	0
2	6,5	0,4	26,9	1
3	6,7	0,5	26,9	1
4	6,5	0,8	27,0	1
5	6,5	0,8	27,0	1
6	6,8	1,0	27,2	1
7	6,8	0,5	27,2	1
8	6,9	0,7	27,4	1
9	6,5	0,4	27,4	1

10	6,5	0,6	27,4	1
11	6,5	0,6	26,9	1
12	6,5	0,9	26,9	1
13	6,3	0,6	26,9	0
14	6,5	0,7	27,2	1
15	6,5	0,5	27,2	1
16	6,5	0,5	27,3	1
17	6,5	0,6	27,3	1
18	6,5	0,6	27,4	1
19	6,3	0,5	27,4	0
20	6,3	0,6	27,4	0
21	6,8	0,8	27,1	1
22	6,7	0,8	27,0	1
23	6,7	0,6	26,9	1
24	6,5	0,7	27,0	1
25	6,8	0,5	27,0	1
26	6,6	0,6	27,2	1
27	6,6	0,8	27,0	1
28	6,5	0,7	27,0	1
29	6,8	0,6	27,2	1
30	6,8	0,6	27,2	1
TOTAL				26

Fuente: Elaboración Propia

$$n_a = n_p = 30$$

Promedio

Reemplazando los valores en la fórmula n=30:

$$\overline{NCFD} = \frac{26}{30}$$

$$\overline{NCFD} = 0.86$$

Varianza:

Reemplazando los valores en la fórmula

$$\sigma_A^2 = 0.138$$

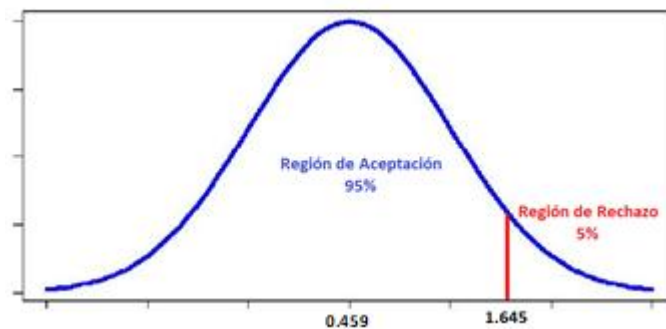
Cálculo de Z

$$Z_c = \frac{\overline{NCFD} - \overline{NCFD}}{\sqrt{\left(\frac{\sigma_A^2}{n_A} + \frac{\sigma_P^2}{n_P}\right)}}$$

$$Z_c = \frac{0.9 - 0.86}{\sqrt{\left(\frac{0.09}{30} + \frac{0.138}{30}\right)}} = 0.459$$

La región crítica para $\alpha = 0.05$, en la tabla distribución Z (**ver Anexo A**) encontramos que $Z_\alpha = 1.645$ Entonces la región crítica de la prueba es: $Z_c = < 0.459, 30 >$

Figura 51 Gráfico de aceptación de Nivel de Completitud Funcional



Fuente: Elaboración Propia

Conclusión

Puesto que: $Z_c = 0.459$ (Z calculado) es menor que $Z_\alpha = 1.645$ (tabular) y estando este valor fuera de la región de rechazo $< 1.645, 30 >$, entonces se acepta H_0 y por consiguiente H_a es rechazada.

Se concluye que el nivel de completitud funcional actual es igual al nivel de completitud funcional con la aplicación móvil, con un nivel de error del 5% y un nivel de confianza del 95%.

6.3.2. Nivel de Corrección Funcional

Definición de Variables

NCFA: Nivel de corrección funcional antes de la implementación de la aplicación móvil propuesta.

NCFD: Nivel de corrección funcional después de la implementación de la aplicación móvil propuesta.

Hipótesis Estadística

H₀: NCFA – NCFD = 0

H_a: NCFA – NCFD > 0

Nivel de Significancia

El nivel de significancia (α) escogido para la prueba de la hipótesis es del 5%. Por lo tanto, el nivel de confianza ($1 - \alpha = 0.95$) será del 95%.

Estadística de Contraste

Puesto que la muestra son las lecturas del pozo seleccionado como muestra, $n=30$, usaremos la distribución normal (**ver Anexo A.**) para lo cual se tiene:

Fórmula N° 1: Promedio

$$\bar{X} = \frac{\sum_{i=1}^n X_i}{n} \dots\dots\dots(1)$$

Fórmula N° 3: Varianza de Z

$$\sigma^2 = \frac{\sum_{i=1}^n (X_i - \bar{X})^2}{n} \dots\dots\dots(2)$$

Fórmula N° 4: Prueba Z

$$Z_c = \frac{\overline{NCFA} - \overline{NCFD}}{\sqrt{\left(\frac{\sigma_A^2}{n_A} + \frac{\sigma_P^2}{n_P}\right)}} \dots\dots\dots(3)$$

Cálculos

Para calcular el nivel de corrección funcional para una muestra de 30 lecturas realizadas para parámetro evaluado es necesario calcular los promedios del nivel de corrección funcional antes y después de la implementación de la aplicación móvil propuesta.

Nivel de corrección funcional respecto a la medición de parámetros de calidad del agua sin la aplicación móvil.

Tabla 24 Nivel de Corrección Funcional sin aplicación móvil

N° de Aciertos para Indicador de Corrección Funcional			
N°	Rangos		Acierto
	pH 6,5 - 8,5	Turbidez 0 - 5	
1	6,9	0,09	1

2	6,8	0,09	1
3	6,8	0,09	1
4	6,8	0,09	1
5	6,9	0,09	1
6	7	0,9	1
7	6,5	0,9	1
8	6,5	3	1
9	6,9	0,24	1
10	6,7	2	1
11	6,8	0,09	1
12	6,4	1,27	0
13	6,4	0,9	0
14	6,9	0,9	1
15	6,7	0,9	1
16	6,7	0,09	1
17	6,5	0,09	1
18	6,3	0,09	0
19	6,8	0,57	1
20	6,8	4,09	1
21	6,5	1,04	1
22	6,9	1,88	1
23	6,8	1,99	1
24	6,9	0,2	1
25	6,6	0,43	1
26	8	0,27	1
27	8,3	0,22	1
28	7,4	0,26	1
29	7,2	0,3	1
30	6,9	0,2	1
TOTAL			27

Fuente: Elaboración Propia

$$n_a = n_p = 30$$

Promedio

Reemplazando los valores en la fórmula $n=30$:

$$\overline{NCF\bar{A}} = \frac{27}{30}$$

$$\overline{NCF\bar{A}} = 0.9$$

Varianza:

Reemplazando los valores en la fórmula

$$\sigma_A^2 = 0.09$$

Nivel de corrección funcional respecto a la medición de parámetros de calidad del agua con la aplicación móvil

Tabla 25 Nivel de Corrección Funcional con aplicación móvil

N° de Aciertos para Indicador de Corrección Funcional				
N°	Rangos			Acierto
	pH 6,5 - 8,5	Turbidez 0 - 5	Temperatura 15 - 35	
1	5,9	0,4	27,0	0
2	6,5	0,4	26,9	1
3	6,7	0,5	26,9	1
4	6,5	0,8	27,0	1
5	6,5	0,8	27,0	1
6	6,8	1,0	27,2	1
7	6,8	0,5	27,2	1
8	6,9	0,7	27,4	1
9	6,5	0,4	27,4	1
10	6,5	0,6	27,4	1
11	6,5	0,6	26,9	1
12	6,5	0,9	26,9	1
13	6,3	0,6	26,9	0
14	6,5	0,7	27,2	1
15	6,5	0,5	27,2	1
16	6,5	0,5	27,3	1
17	6,5	0,6	27,3	1
18	6,5	0,6	27,4	1
19	6,3	0,5	27,4	0
20	6,3	0,6	27,4	0
21	6,8	0,8	27,1	1
22	6,7	0,8	27,0	1
23	6,7	0,6	26,9	1
24	6,5	0,7	27,0	1
25	6,8	0,5	27,0	1
26	6,6	0,6	27,2	1
27	6,6	0,8	27,0	1
28	6,5	0,7	27,0	1
29	6,8	0,6	27,2	1

30	6,8	0,6	27,2	1
TOTAL				26

Fuente: Elaboración Propia

$$n_a = n_p = 30$$

Promedio

Reemplazando los valores en la fórmula $n=30$:

$$\overline{NCFD} = \frac{26}{30}$$

$$\overline{NCFD} = 0.86$$

Varianza:

Reemplazando los valores en la fórmula

$$\sigma_A^2 = 0.138$$

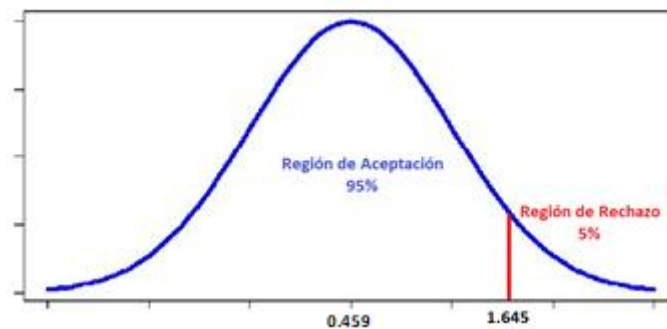
Cálculo de Z

$$Z_c = \frac{\overline{NCFA} - \overline{NCFD}}{\sqrt{\left(\frac{\sigma_A^2}{n_A} + \frac{\sigma_P^2}{n_P}\right)}}$$

$$Z_c = \frac{0.9 - 0.86}{\sqrt{\left(\frac{0.09}{30} + \frac{0.138}{30}\right)}} = 0.459$$

La región crítica para $\alpha = 0.05$, en la tabla distribución Z (**ver Anexo A**) encontramos que $Z_\alpha = 1.645$ Entonces la región crítica de la prueba es: $Z_c = < 0.459, 30 >$

Figura 52 Gráfico de Aceptación de Nivel de Corrección Funcional



Fuente: Elaboración Propia

Conclusión

Puesto que: $Z_c = 0.459$ (Z calculado) es menor que $Z_\alpha = 1.645$ (tabular) y estando este valor fuera de la región de rechazo $< 1.645, 30 >$, entonces se acepta H_0 y por consiguiente H_a es rechazada.

Se concluye que el nivel de corrección funcional actual es igual al nivel de corrección funcional con la aplicación móvil, con un nivel de error del 5% y un nivel de confianza del 95%.

6.3.3. Nivel de Comportamiento Temporal

Definición de Variables

NCTA: Nivel de comportamiento temporal antes de la implementación de la aplicación móvil propuesta.

NCTD: Nivel de comportamiento temporal después de la implementación de la aplicación móvil propuesta.

Hipótesis Estadística

H₀: NCTA – NCTD < 0

H_a: NCTA – NCTD > 0

Nivel de Significancia

El nivel de significancia (α) escogido para la prueba de la hipótesis es del 5%. Por lo tanto, el nivel de confianza ($1 - \alpha = 0.95$) será del 95%.

Estadística de Contraste

Puesto que la muestra son las lecturas del pozo seleccionado como muestra, $n=30$, usaremos la distribución normal (**ver Anexo A.**) para lo cual se tiene:

Fórmula N° 1: Promedio

$$\bar{X} = \frac{\sum_{i=1}^n X_i}{n} \dots\dots\dots(1)$$

Fórmula N° 5: Varianza de Z

$$\sigma^2 = \frac{\sum_{i=1}^n (X_i - \bar{X})^2}{n} \dots\dots\dots(2)$$

Fórmula N° 6: Prueba Z

$$Z_c = \frac{\overline{NCTA} - \overline{NCTD}}{\sqrt{\left(\frac{\sigma_A^2}{n_A} + \frac{\sigma_P^2}{n_P}\right)}} \dots\dots\dots(3)$$

Cálculos

Para calcular el nivel de comportamiento temporal para una muestra de 30 lecturas realizadas para parámetro evaluado es necesario calcular los promedios del nivel de comportamiento temporal antes y después de la implementación de la aplicación móvil propuesta.

Nivel de comportamiento temporal respecto a la medición de parámetros de calidad del agua sin la aplicación móvil.

Tabla 26 Nivel del comportamiento temporal sin la aplicación móvil.

Parámetro	Tiempo (s.)
L1	1800
L2	2100
L3	2400
L4	2400
L5	2220
L6	1980
L7	2160
L8	2100
L9	2100
L10	1980
L11	2400
L12	2280
L13	2400
L14	2100
L15	2100

L16	1800
L17	2160
L18	2220
L19	2400
L20	2100
L21	2100
L22	2220
L23	1980
L24	2100
L25	2280
L26	2280
L27	2100
L28	2160
L29	1980
L30	2280
Total	64680
Promedio	2156

Fuente: Elaboración Propia

Nivel de comportamiento temporal respecto a la medición de parámetros de calidad del agua con la aplicación móvil

Tabla 27 Nivel de comportamiento temporal con la aplicación móvil

Parámetro	Tiempo(s.)
L1	0,3
L2	0,3
L3	0,4
L4	0,4
L5	0,4
L6	0,4
L7	0,3
L8	0,1
L9	0,2
L10	0,2
L11	0,4
L12	0,3
L13	0,4
L14	0,2
L15	0,2
L16	0,3
L17	0,2
L18	0,2

L19	0,3
L20	0,3
L21	0,2
L22	0,4
L23	0,3
L24	0,3
L25	0,2
L26	0,2
L27	0,3
L28	0,4
L29	0,3
L30	0,3
Total	8,7
Promedio	0,29

Fuente: Elaboración Propia

Cálculo de Varianza

Tabla 28 Cálculo de Varianza antes de la implementación de aplicación móvil

Cálculo de Varianza antes de implementación de aplicación móvil	
	Tiempo (s)
Promedio (\bar{X})	2156.0
Varianza (σ^2)	25904

Fuente: Elaboración Propia

Tabla 29 Cálculo de Varianza después de implementación de aplicación móvil

Cálculo de Varianza después de implementación de aplicación móvil	
	Tiempo (s)
Promedio (\bar{X})	0.3
Varianza (σ^2)	0.007

Fuente: Elaboración Propia

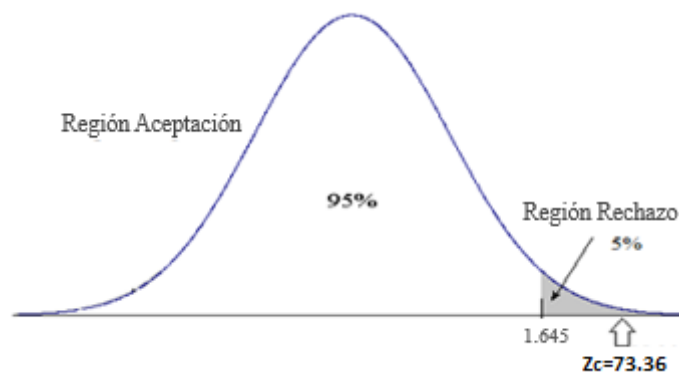
Cálculo de Z

$$Z_c = \frac{\overline{\text{NCTA}} - \text{NCTD}}{\sqrt{\left(\frac{\sigma_A^2}{n_A} + \frac{\sigma_P^2}{n_P}\right)}}$$

$$Z_c = \frac{2156 - 0.3}{\sqrt{\left(\frac{25904}{30} + \frac{0.007}{30}\right)}} = 73.36$$

La región crítica para $\alpha = 0.05$, en la tabla distribución Z (**ver Anexo A**) encontramos que $Z_\alpha = 1.645$ Entonces la región crítica de la prueba es: $Z_c = < 73.36, 30 >$

Figura 53 Gráfico de Aceptación de Nivel de Comportamiento Temporal



Fuente: Elaboración Propia

Puesto que: $Z_c = 73.36$ (Z calculado) es mayor que $Z_\alpha = 1.645$ (tabular) y estando este valor en la región de rechazo $< 1.645, 30 >$, entonces se rechaza H_0 y por consiguiente H_a es aceptada. Se concluye que el Tiempo de medición de parámetros de calidad del agua con la aplicación móvil, es menor que el tiempo promedio que demoran en realizar las mediciones de manera manual, con un nivel de error del 5% y un nivel de confianza del 95%.

CAPÍTULO 7. DISCUSIÓN

7.1. Indicador Cuantitativo “Compleitud Funcional”.

De acuerdo a la estimación de los promedios de aciertos en niveles de Compleitud Funcional con el proceso actual y el sistema propuesto, los cuales se calcularon anteriormente, teniendo como promedio de aciertos el 100%, como base para los cálculos.

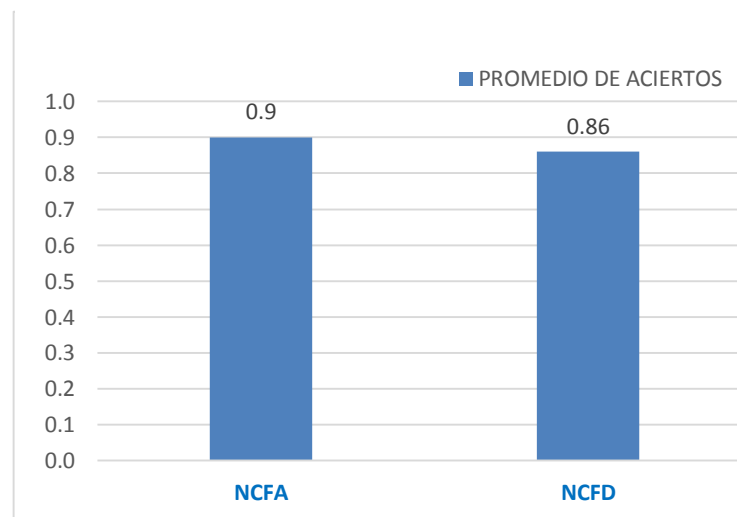
Tabla 30 Nivel de Impacto – Indicador 1

NCFD		NCFD		Nivel Impacto	
Promedio de Aciertos	Porcentaje (%)	Promedio de Aciertos	Porcentaje (%)	Aciertos	Porcentaje (%)
0.9	90%	0.86	86%	0.04	4%

Fuente: Elaboración Propia

Se puede observar que el nivel de Compleitud Funcional promedio de aciertos es 0.9 y con la implantación del sistema es de 0.86, teniendo un nivel de impacto de decremento del 4%; es decir, que existe una diferencia mínima con respecto a los resultados obtenidos antes y después de aplicar la aplicación propuesta, como podemos apreciar en la siguiente gráfica:

Figura 54 Nivel de Compleitud Funcional



Fuente: Elaboración Propia

7.2. Indicador Cuantitativo “Nivel de Corrección Funcional”.

De acuerdo a la estimación de los promedios de aciertos en niveles de Corrección Funcional con el proceso actual y el sistema propuesto, los cuales se calculó anteriormente, teniendo como promedio de aciertos el 100%, como base para los cálculos.

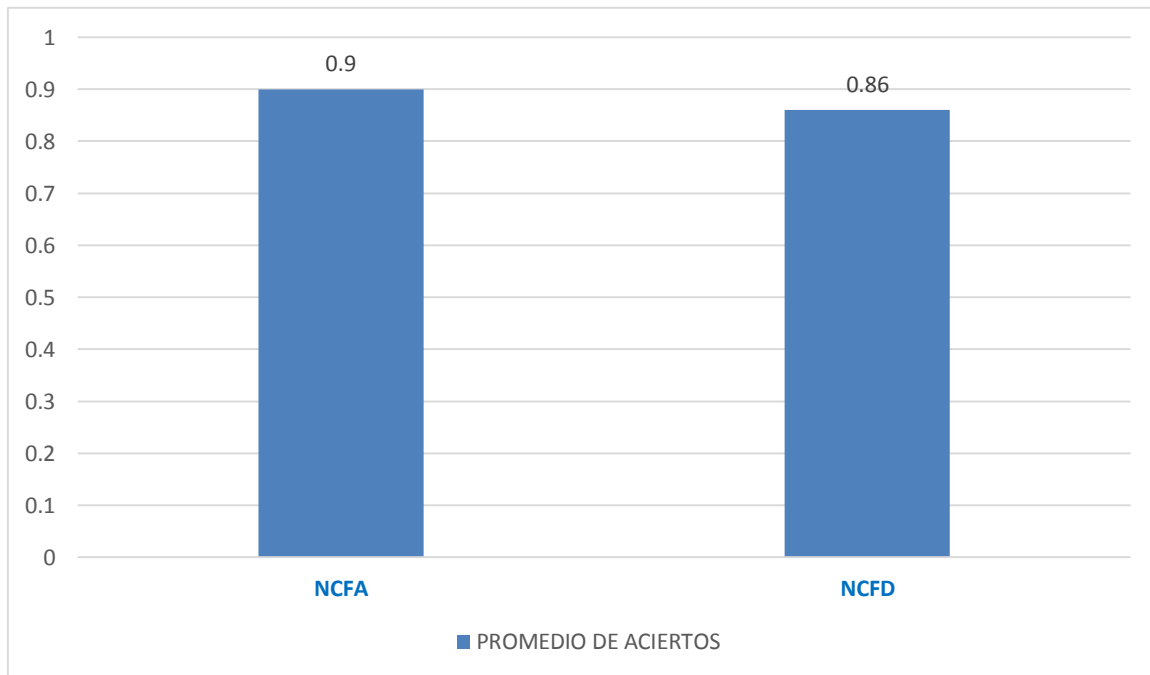
Tabla 31 Nivel de Impacto – Indicador 2

NCFA		NCFD		Nivel Impacto	
Promedio de Aciertos	Porcentaje (%)	Promedio de Aciertos	Porcentaje (%)	Aciertos	Porcentaje (%)
0.9	90%	0.86	86%	0.04	4%

Fuente: Elaboración Propia

Se puede observar que el nivel de Completitud Funcional promedio de aciertos es 0.9 y con la implantación del sistema es de 0.86, teniendo un nivel de impacto de decremento del 4%; es decir, que existe una diferencia mínima con respecto a los resultados obtenidos antes y después de aplicar la aplicación propuesta, como podemos apreciar en la siguiente gráfica:

Figura 55 Nivel de Corrección Funcional



Fuente: Elaboración Propia

7.3. Indicador Cuantitativo “Comportamiento Temporal”

De acuerdo a la estimación de los tiempos promedios de obtención de lecturas de calidad agua con el proceso actual y el sistema propuesto valores previamente calculados. Tomando como base los cálculos de tiempo en obtener lecturas de datos de calidad de agua como el 100%.

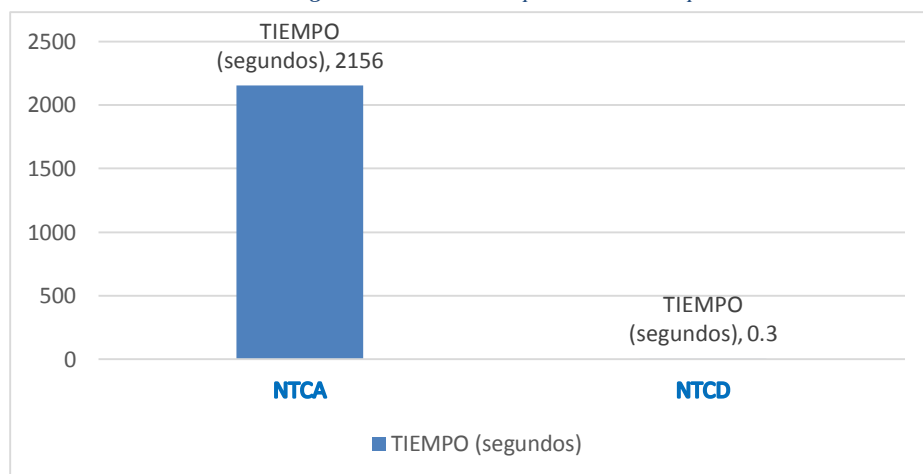
Tabla 32 Nivel de Comportamiento Temporal

NCTA		NCTD		Nivel Impacto: Decremento	
Puntaje (segundos)	Porcentaje (%)	Puntaje (segundos)	Porcentaje (%)	Puntaje (segundos)	Porcentaje (%)
2156.0	100%	0.3	0.01%	2155.7	99.99%

Fuente: Elaboración Propia

Se puede observar que el tiempo promedio de obtención de lecturas de datos de calidad de agua en el proceso actual es de 2156 segundos y con la implantación del sistema propuesto es de 0.3 segundos, teniendo un nivel de impacto de decremento 2155.7 segundos equivalentes al 99.99% como podemos apreciar en la siguiente gráfica:

Figura 56 Nivel de Comportamiento Temporal



Fuente: Elaboración Propia

7.4. Indicador Cualitativo “Eficacia de los Resultados”

En concordancia con los resultados obtenidos para los indicadores antes mencionados y evaluados, y verificando que estos van acorde al Reglamento de Calidad de Agua Potable que establece el Estado Peruano, el Jefe del ATM Ing. Juan Manuel Liñán Rodríguez, da validez de que los valores obtenidos por el sistema propuesto, planteado en el presente trabajo, cumple eficazmente con la finalidad propuesta. **(Ver Anexo C)**

CONCLUSIONES

1. Se concluye que el nivel de Completitud Funcional actual tiene gran similitud al nivel de Completitud Funcional con la aplicación móvil, con una diferencia de 4% con respecto a los resultados medidos.
2. El tiempo promedio de obtención de las lecturas de datos de calidad de agua en el proceso actual es de 2156 segundos (100%) y con la implantación del sistema propuesto es de 0.3 segundos (0.01 %), teniendo un nivel de impacto de decremento 2155.7 segundos equivalentes al 99.99%.
3. Se concluye que el nivel de corrección funcional actual es igual al nivel de corrección funcional con la aplicación móvil, con un nivel de error del 5% y un nivel de confianza del 95%, con una diferencia de 10% con respecto a los resultados medidos.
4. Se concluye que el sistema propuesto alcanzó el nivel de eficacia en base a la evaluación realiza por el especialista del área.

RECOMENDACIONES

Para mejorar la continuidad operacional de sistema propuesto, se presenta las siguientes recomendaciones:

1. Se recomienda implementar el sistema propuesto en todos los pozos de agua de la región para garantizar la calidad del agua a los usuarios de la región en tiempo real.
2. Es importante que las entidades y usuarios finales del sistema propuesto reciban capacitaciones para su uso correcto y aprovechamiento de las diferentes opciones del aplicativo móvil.
3. Se deben realizar calibración y mantenimiento de los sensores en los pozos de agua al visualizar mediciones anormales, de manera que se garantice la precisión de las lecturas de datos de la calidad de agua.
4. Se recomienda como posible mejora implementar el sensor de cloro residual, pues permitiría controlar en tiempo real el nivel de cloro en el pozo de agua.

REFERENCIAS

Montañez, E. (26 de Julio de 2017) El agua del grifo, de calidad en el 90% de los municipios. Recuperado de http://www.abc.es/sociedad/abci-agua-grifo-calidad-90-por-ciento-municipios-201707252137_noticia.html

Anónimo (31 de abril de 2016) ¿Sabe usted dónde se toma la mejor y la peor agua del país? *El Tiempo*. Recuperado de <http://www.eltiempo.com/multimedia/especiales/calidad-del-agua-en-colombia/16555634/1/index.html>

Anónimo (15 de noviembre de 2017). Laredo: 66 % de sistemas de agua potable y alcantarillado están deterioradas. *Correo*. Recuperado de <https://diariocorreo.pe/edicion/la-libertad/laredo-66-de-sistemas-de-agua-potable-y-alcantarillado-estan-deterioradas-786375/>

Anónimo (25 de Julio de 2018). Desarrollan proyecto que ayuda evaluar calidad de agua en colegios de Laredo. *Día a Día Trujillo*. Recuperado de <http://diaadiatrujillo.com/2018/07/25/desarrollan-proyecto-que-ayuda-evaluar-calidad-de-agua-en-colegios-de-laredo/#>.

Superintendencia Nacional de Servicio y Saneamiento (2004). Calidad del Agua Potable en el Perú (p.45). Lima: Tareas Gráficas.

Organización Mundial de la Salud (2006). Guías para la Calidad del Agua Potable (Vol. 1, p.89). Suiza.

Ministerio de trabajo y asuntos Sociales (1999). NTP 538: Legionelosis: medidas de prevención y control en instalaciones de suministro de agua (p.1). España.

Digesa (2006). Parámetros Físicoquímicos: Temperatura (p. 9).

Digesa (2006). Parámetros Físicoquímicos: Potencial de Hidrógeno (p.7).

Digesa (2006). Parámetros Físicoquímicos: Electro conductividad (p.18).

Digesa (2006). Parámetros Físicoquímicos: Turbidez, (p.19).

Arduino (2018). Comparación de las especificaciones de tarjetas de desarrollo. Recuperado de <https://www.arduino.cc/en/Products/Compare>

Ionicframework (2018). Conceptos básicos. Recuperado de <https://ionicframework.com/docs/intro/concepts/>

Esaú A. (2016). Ionic Framework, ventajas y desventajas. Recuperado de <https://openwebinars.net/blog/ionic-framework-ventajas-desventajas/>

Electrocrea (2014). Temperatura con Termistor NTC 10K y Arduino. Recuperado de <https://www.instructables.com/id/Temperatura-con-Termistor-NTC-10k-y-Arduino/>

DFRobot (2017). Sensor de turbidez SKU: SEN0189. Recuperado de https://www.dfrobot.com/wiki/index.php/Turbidity_sensor_SKU:_SEN0189

DFRobot (2017). Sensor de turbidez SKU SEN0189: Diagrama de Conexión. Recuperado de https://www.dfrobot.com/wiki/index.php/Turbidity_sensor_SKU:_SEN0189

DFRobot (2017). Sensor de Conductividad Eléctrica DFR0300. Recuperado de https://www.dfrobot.com/wiki/index.php/Analog_EC_Meter_SKU:DFR0300

DFRobot (2017). Sensor de Conductividad Eléctrica DFR0300: Diagrama de Conexión. Recuperado de https://www.dfrobot.com/wiki/index.php/Analog_EC_Meter_SKU:DFR0300

DFRobot (2017). Medidor de pH (SKU: SEN0161). Recuperado de [https://www.dfrobot.com/wiki/index.php/PH_meter\(SKU:_SEN0161\)](https://www.dfrobot.com/wiki/index.php/PH_meter(SKU:_SEN0161))

DFRobot (2017). Medidor de pH (SKU: SEN0161): Diagrama de Conexión. Recuperado de [https://www.dfrobot.com/wiki/index.php/PH_meter\(SKU:_SEN0161\)](https://www.dfrobot.com/wiki/index.php/PH_meter(SKU:_SEN0161))

Jacobson, I. (1999). El Proceso Unificado de desarrollo de software. Inglaterra: Cambridge University Press.

Alistair Cockburn (2006). Agile Software Development: The Cooperative Game. (2.^a ed.).

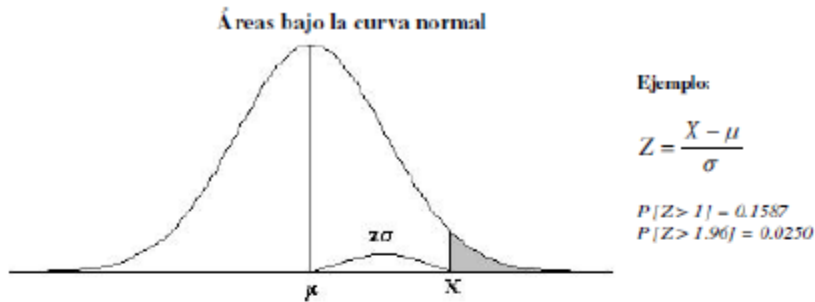
Ecured (2018). Programación Extrema (XP). Recuperado de [https://www.ecured.cu/Programaci%C3%B3n_Extrema_\(XP\)](https://www.ecured.cu/Programaci%C3%B3n_Extrema_(XP))

Ecured (2018). Programación Extrema (XP). Recuperado de <https://www.ecured.cu/SCRUM>

Malpica Velásquez, C. (2014). *Aplicación de la metodología SCRUM para incrementar la productividad del proceso de desarrollo de software en la empresa CCJ S.A.C. Lima.* Universidad Nacional del Centro del Perú, Huancayo, Perú.

ANEXOS

ANEXO A. Distribución Normal



Desv. normal x	0.00	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09
0.0	0.5000	0.4960	0.4920	0.4880	0.4840	0.4801	0.4761	0.4721	0.4681	0.4641
0.1	0.4602	0.4562	0.4522	0.4483	0.4443	0.4404	0.4364	0.4325	0.4286	0.4247
0.2	0.4207	0.4168	0.4129	0.4090	0.4052	0.4013	0.3974	0.3936	0.3897	0.3859
0.3	0.3821	0.3783	0.3745	0.3707	0.3669	0.3632	0.3594	0.3557	0.3520	0.3483
0.4	0.3446	0.3409	0.3372	0.3336	0.3300	0.3264	0.3228	0.3192	0.3156	0.3121
0.5	0.3085	0.3050	0.3015	0.2981	0.2946	0.2912	0.2877	0.2843	0.2810	0.2776
0.6	0.2743	0.2709	0.2676	0.2643	0.2611	0.2578	0.2546	0.2514	0.2483	0.2451
0.7	0.2420	0.2389	0.2358	0.2327	0.2296	0.2266	0.2236	0.2206	0.2177	0.2148
0.8	0.2119	0.2090	0.2061	0.2033	0.2005	0.1977	0.1949	0.1922	0.1894	0.1867
0.9	0.1841	0.1814	0.1788	0.1762	0.1736	0.1711	0.1685	0.1660	0.1635	0.1611
1.0	0.1587	0.1562	0.1539	0.1515	0.1492	0.1469	0.1446	0.1423	0.1401	0.1379
1.1	0.1357	0.1335	0.1314	0.1292	0.1271	0.1251	0.1230	0.1210	0.1190	0.1170
1.2	0.1151	0.1131	0.1112	0.1093	0.1075	0.1056	0.1038	0.1020	0.1003	0.0985
1.3	0.0968	0.0951	0.0934	0.0918	0.0901	0.0885	0.0869	0.0853	0.0838	0.0823
1.4	0.0808	0.0793	0.0778	0.0764	0.0749	0.0735	0.0721	0.0708	0.0694	0.0681
1.5	0.0668	0.0655	0.0643	0.0630	0.0618	0.0606	0.0594	0.0582	0.0571	0.0559
1.6	0.0548	0.0537	0.0526	0.0516	0.0505	0.0495	0.0485	0.0475	0.0465	0.0455
1.7	0.0446	0.0436	0.0427	0.0418	0.0409	0.0401	0.0392	0.0384	0.0375	0.0367
1.8	0.0359	0.0351	0.0344	0.0336	0.0329	0.0322	0.0314	0.0307	0.0301	0.0294
1.9	0.0287	0.0281	0.0274	0.0268	0.0262	0.0256	0.0250	0.0244	0.0239	0.0233
2.0	0.0228	0.0222	0.0217	0.0212	0.0207	0.0202	0.0197	0.0192	0.0188	0.0183
2.1	0.0179	0.0174	0.0170	0.0166	0.0162	0.0158	0.0154	0.0150	0.0146	0.0143
2.2	0.0139	0.0136	0.0132	0.0129	0.0125	0.0122	0.0119	0.0116	0.0113	0.0110
2.3	0.0107	0.0104	0.0102	0.0099	0.0096	0.0094	0.0091	0.0089	0.0087	0.0084
2.4	0.0082	0.0080	0.0078	0.0075	0.0073	0.0071	0.0069	0.0068	0.0066	0.0064
2.5	0.0062	0.0060	0.0059	0.0057	0.0055	0.0054	0.0052	0.0051	0.0049	0.0048
2.6	0.0047	0.0045	0.0044	0.0043	0.0041	0.0040	0.0039	0.0038	0.0037	0.0036
2.7	0.0035	0.0034	0.0033	0.0032	0.0031	0.0030	0.0029	0.0028	0.0027	0.0026
2.8	0.0026	0.0025	0.0024	0.0023	0.0023	0.0022	0.0021	0.0021	0.0020	0.0019
2.9	0.0019	0.0018	0.0018	0.0017	0.0016	0.0016	0.0015	0.0015	0.0014	0.0014
3.0	0.0013	0.0013	0.0013	0.0012	0.0012	0.0011	0.0011	0.0011	0.0010	0.0010

ANEXO B. Lecturas Pre Test Ilevadas a cabo por ATM de Laredo

UBIGEO_CP	NOMBRE_CP	PROVINCIA_CP	DISTRITO_CP	Nitritos (mg/L)	Potencial de Hidrógeno (pH)	Sólidos Totales disueltos (mg/L)	Sulfatos (mg/L)	Turbiedad (UNT)	Cloruros (mg/L)	Aluminio (mg/L)	Antimonio (mg/L)	Arsénico (mg/L)	Bario (mg/L)	Berilio (mg/L)	Boro (mg/L)	Cadmio (mg/L)	Cobre (mg/L)	Cromo total (mg/L)	Hierro (mg/L)	Manganeso (mg/L)	Mercurio (mg/L)	M
1,3E+09	BELLO HORIZONT	TRUJILLO	LAREDO		6,9		1,11	0,09		0,0046		0,006				0,0023	0,0018	0,002	0,003	0,0077	0,0009	
1,3E+09	BELLO HORIZONT	TRUJILLO	LAREDO		6,8		0,83	0,09		0,0046		0,006				0,0023	0,0018	0,002	0,003	0,0077	0,0009	
1,3E+09	BELLO HORIZONT	TRUJILLO	LAREDO		6,8		0,83	0,09		0,0047		0,006				0,0023	0,0018	0,002	0,003	0,0077	0,0009	
1,3E+09	BELLO HORIZONT	TRUJILLO	LAREDO		6,8		0,96	0,09		0,0047		0,0061				0,0023	0,0018	0,002	0,003	0,0077	0,001	
1,3E+09	BELLO HORIZONT	TRUJILLO	LAREDO		6,9		4,95	0,09		0,0046		0,006				0,0023	0,0018	0,002	0,09	0,0077	0,0009	
1,3E+09	BELLO HORIZONT	TRUJILLO	LAREDO		7		1,28	0,9		0,0046		0,006				0,0023	0,0018	0,002	0,062	0,0077	0,0009	
1,3E+09	BELLO HORIZONT	TRUJILLO	LAREDO		6,5		3,99	0,9		0,0046		0,006				0,0023	0,0018	0,002	0,077	0,0077	0,0009	
1,3E+09	BELLO HORIZONT	TRUJILLO	LAREDO		6,5		0,9	3		0,0046		0,006				0,0023	0,0018	0,002	0,002	0,018	0,0009	
1,3E+09	BELLO HORIZONT	TRUJILLO	LAREDO		6,9			0,24		0,17	0,01	0,0009	0,011	0,0001	0,17	0,0004	0,009	0,0009	0,039	0,001		
1,3E+09	BELLO HORIZONT	TRUJILLO	LAREDO		6,7		66,94	2		0,0046		0,006				0,0023	0,0018	0,002	0,069	0,012	0,0009	
1,3E+09	BELLO HORIZONT	TRUJILLO	LAREDO		6,8		151,1	0,09		0,0046		0,006				0,0023	0,0018	0,0018	0,002	0,018	0,0009	
1,3E+09	BELLO HORIZONT	TRUJILLO	LAREDO		6,4		50,18	1,27		0,0046		0,006				0,0023	0,0018	0,002	0,073	0,0077	0,0009	
1,3E+09	BELLO HORIZONT	TRUJILLO	LAREDO	0,002	6,4		3,76	0,9		0,0046	0,0057	0,006	0,0015	0,0026	0,11	0,0023	0,0018	0,002	0,039	0,0077	0,0009	
1,3E+09	BELLO HORIZONT	TRUJILLO	LAREDO	0,002	6,9	64	5,02	0,9		0,0046	0,0057	0,006	0,0015	0,0026	0,193	0,0023	0,0018	0,002	0,065	0,0077	0,0009	
1,3E+09	BELLO HORIZONT	TRUJILLO	LAREDO	0,002	6,7		5,06	0,9		0,0046	0,0057	0,006	0,0015	0,0026	0,128	0,0023	0,0018	0,002	0,112	0,0077	0,0009	
1,3E+09	BELLO HORIZONT	TRUJILLO	LAREDO		6,7		36,9	0,09		0,0046		0,006				0,0023	0,008	0,002	0,015	0,0077	0,0009	
1,3E+09	BELLO HORIZONT	TRUJILLO	LAREDO		6,5		80,42	0,09		0,016		0,006				0,0023	0,0018	0,002	0,08	0,0077	0,001	
1,3E+09	BELLO HORIZONT	TRUJILLO	LAREDO		6,3		6,2	0,09		0,034		0,006				0,0023	0,007	0,002	0,01	0,0077	0,0009	
1,3E+09	BELLO HORIZONT	TRUJILLO	LAREDO		6,8		5,29	0,57		0,0046		0,006				0,0023	0,0018	0,002	11,26	0,125	0,0009	
1,3E+09	BELLO HORIZONT	TRUJILLO	LAREDO		6,8		3,43	4,09		0,033		0,006				0,0023	0,0018	0,002	0,022	0,0077	0,0009	
1,3E+09	BELLO HORIZONT	TRUJILLO	LAREDO		6,5		1,75	1,04		0,04		0,006				0,0023	0,0018	0,002	0,028	0,0077	0,0009	
1,3E+09	BELLO HORIZONT	TRUJILLO	LAREDO		6,9		1,78	1,88		0,037		0,006				0,0023	0,0018	0,002	0,034	0,0077	0,0009	
1,3E+09	BELLO HORIZONT	TRUJILLO	LAREDO		6,8		2,16	1,99		0,108		0,006				0,0023	0,002	0,002	0,361	0,0077	0,0009	
1,3E+09	BELLO HORIZONT	TRUJILLO	LAREDO		6,9		167,9	0,2		0,0046		0,006				0,0023	0,0018	0,002	0,003	0,0077	0,0009	
1,3E+09	BELLO HORIZONT	TRUJILLO	LAREDO		6,6		18,63	0,43		0,135		0,006				0,0023	0,0018	0,002	0,055	0,0077	0,0009	
1,3E+09	BELLO HORIZONT	TRUJILLO	LAREDO		8		224,3	0,27		0,0046		0,006				0,0023	0,0018	0,002	0,005	0,0077	0,0009	
1,3E+09	BELLO HORIZONT	TRUJILLO	LAREDO		8,3		225,5	0,22		0,0046		0,006				0,0023	0,0018	0,002	0,011	0,029	0,0009	
1,3E+09	BELLO HORIZONT	TRUJILLO	LAREDO		7,4		217,5	0,26		0,0046		0,0061				0,0023	0,0018	0,002	0,007	0,0077	0,0009	
1,3E+09	BELLO HORIZONT	TRUJILLO	LAREDO		7,2		239	0,3		0,0046		0,006				0,0023	0,0018	0,002	0,003	0,0077	0,0009	
1,3E+09	BELLO HORIZONT	TRUJILLO	LAREDO		6,9		270,8	0,2		0,0046		0,006				0,0023	0,0018	0,002	0,003	0,0077	0,0009	
1,3E+09	BELLO HORIZONT	TRUJILLO	LAREDO		7,45		217,5	0,26		0,0046		0,0061				0,0023	0,0018	0,002	0,007	0,0077	0,0009	
1,3E+09	BELLO HORIZONT	TRUJILLO	LAREDO		7,57		239	0,3		0,0046		0,006				0,0023	0,0018	0,002	0,003	0,0077	0,0009	
1,3E+09	BELLO HORIZONT	TRUJILLO	LAREDO		7,65		270,8	0,2		0,0046		0,006				0,0023	0,0018	0,002	0,003	0,0077	0,0009	

ANEXO B. Lecturas Pre Test llevadas a cabo por ATM de Laredo

UBIGEO _CP	NOMBRE_CP	PROVINCIA_CP	DISTRITO_CP	Nitritos (mg/L)	Potenci al de Hidróge no (pH)	Sólidos Totales disueltos (mg/L)	Sulfatos (mg/L)	Turbieda d (UNT)	Cloruros (mg/L)	Aluminio (mg/L)	Antimoni o (mg/L)	Arsénico (mg/L)	Bario (mg/L)	Berilio (mg/L)	Boro (mg/L)	Cadmio (mg/L)	Cobre (mg/L)	Cromo total (mg/L)	Hierro (mg/L)	Mangan eso (mg/L)	Mercurio (mg/L)	M
1,3E+09	BELLO HORIZONT TRUJILLO	LAREDO			6,9		1,11	0,09		0,0046		0,006				0,0023	0,0018	0,002	0,003	0,0077	0,0009	
1,3E+09	BELLO HORIZONT TRUJILLO	LAREDO			6,8		0,83	0,09		0,0046		0,006				0,0023	0,0018	0,002	0,003	0,0077	0,0009	
1,3E+09	BELLO HORIZONT TRUJILLO	LAREDO			6,8		0,83	0,09		0,0047		0,006				0,0023	0,0018	0,002	0,003	0,0077	0,0009	
1,3E+09	BELLO HORIZONT TRUJILLO	LAREDO			6,8		0,96	0,09		0,0047		0,0061				0,0023	0,0018	0,002	0,003	0,0077	0,001	
1,3E+09	BELLO HORIZONT TRUJILLO	LAREDO			6,9		4,95	0,09		0,0046		0,006				0,0023	0,0018	0,002	0,09	0,0077	0,0009	
1,3E+09	BELLO HORIZONT TRUJILLO	LAREDO			7		1,28	0,9		0,0046		0,006				0,0023	0,0018	0,002	0,062	0,0077	0,0009	
1,3E+09	BELLO HORIZONT TRUJILLO	LAREDO			6,5		3,99	0,9		0,0046		0,006				0,0023	0,0018	0,002	0,077	0,0077	0,0009	
1,3E+09	BELLO HORIZONT TRUJILLO	LAREDO			6,5		0,9	3		0,0046		0,006				0,0023	0,0018	0,002	0,002	0,018	0,0009	
1,3E+09	BELLO HORIZONT TRUJILLO	LAREDO			6,9			0,24		0,17	0,01	0,0009	0,011	0,0001	0,17	0,0004	0,009	0,0009	0,039	0,001		
1,3E+09	BELLO HORIZONT TRUJILLO	LAREDO			6,7		66,94	2		0,0046		0,006				0,0023	0,0018	0,002	0,069	0,012	0,0009	
1,3E+09	BELLO HORIZONT TRUJILLO	LAREDO			6,8		15,1	0,09		0,0046		0,006				0,0023	0,0018	0,0018	0,002	0,018	0,0009	
1,3E+09	BELLO HORIZONT TRUJILLO	LAREDO			6,4		50,18	1,27		0,0046		0,006				0,0023	0,0018	0,002	0,073	0,0077	0,0009	
1,3E+09	BELLO HORIZONT TRUJILLO	LAREDO		0,002	6,4		3,76	0,9		0,0046	0,0057	0,006	0,0015	0,0026	0,11	0,0023	0,0018	0,002	0,099	0,0077	0,0009	
1,3E+09	BELLO HORIZONT TRUJILLO	LAREDO		0,002	6,9	64	5,02	0,9		0,0046	0,0057	0,006	0,0015	0,0026	0,193	0,0023	0,0018	0,002	0,065	0,0077	0,0009	
1,3E+09	BELLO HORIZONT TRUJILLO	LAREDO		0,002	6,7		5,06	0,9		0,0046	0,0057	0,006	0,0015	0,0026	0,128	0,0023	0,0018	0,002	0,112	0,0077	0,0009	
1,3E+09	BELLO HORIZONT TRUJILLO	LAREDO			6,7		36,9	0,09		0,0046		0,006				0,0023	0,008	0,002	0,015	0,0077	0,0009	
1,3E+09	BELLO HORIZONT TRUJILLO	LAREDO			6,5		80,42	0,09		0,016		0,006				0,0023	0,0018	0,002	0,08	0,0077	0,001	
1,3E+09	BELLO HORIZONT TRUJILLO	LAREDO			6,3		6,2	0,09		0,034		0,006				0,0023	0,007	0,002	0,01	0,0077	0,0009	
1,3E+09	BELLO HORIZONT TRUJILLO	LAREDO			6,8		5,29	0,57		0,0046		0,006				0,0023	0,0018	0,002	11,26	0,125	0,0009	
1,3E+09	BELLO HORIZONT TRUJILLO	LAREDO			6,8		3,43	4,09		0,033		0,006				0,0023	0,0018	0,002	0,022	0,0077	0,0009	
1,3E+09	BELLO HORIZONT TRUJILLO	LAREDO			6,5		1,75	1,04		0,04		0,006				0,0023	0,0018	0,002	0,028	0,0077	0,0009	
1,3E+09	BELLO HORIZONT TRUJILLO	LAREDO			6,9		1,78	1,88		0,037		0,006				0,0023	0,0018	0,002	0,094	0,0077	0,0009	
1,3E+09	BELLO HORIZONT TRUJILLO	LAREDO			6,8		2,16	1,99		0,108		0,006				0,0023	0,002	0,002	0,361	0,0077	0,0009	
1,3E+09	BELLO HORIZONT TRUJILLO	LAREDO			6,9		167,9	0,2		0,0046		0,006				0,0023	0,0018	0,002	0,003	0,0077	0,0009	
1,3E+09	BELLO HORIZONT TRUJILLO	LAREDO			6,6		18,63	0,43		0,135		0,006				0,0023	0,0018	0,002	0,055	0,0077	0,0009	
1,3E+09	BELLO HORIZONT TRUJILLO	LAREDO			8		224,3	0,27		0,0046		0,006				0,0023	0,0018	0,002	0,005	0,0077	0,0009	
1,3E+09	BELLO HORIZONT TRUJILLO	LAREDO			8,3		225,5	0,22		0,0046		0,006				0,0023	0,0018	0,002	0,011	0,029	0,0009	
1,3E+09	BELLO HORIZONT TRUJILLO	LAREDO			7,4		217,5	0,26		0,0046		0,0061				0,0023	0,0018	0,002	0,007	0,0077	0,0009	
1,3E+09	BELLO HORIZONT TRUJILLO	LAREDO			7,2		239	0,3		0,0046		0,006				0,0023	0,0018	0,002	0,003	0,0077	0,0009	
1,3E+09	BELLO HORIZONT TRUJILLO	LAREDO			6,9		270,8	0,2		0,0046		0,006				0,0023	0,0018	0,002	0,003	0,0077	0,0009	
1,3E+09	BELLO HORIZONT TRUJILLO	LAREDO			7,45		217,5	0,26		0,0046		0,0061				0,0023	0,0018	0,002	0,007	0,0077	0,0009	
1,3E+09	BELLO HORIZONT TRUJILLO	LAREDO			7,57		239	0,3		0,0046		0,006				0,0023	0,0018	0,002	0,003	0,0077	0,0009	
1,3E+09	BELLO HORIZONT TRUJILLO	LAREDO			7,65		270,8	0,2		0,0046		0,006				0,0023	0,0018	0,002	0,003	0,0077	0,0009	



ANEXO C. Constancia de Validez de Resultados

CONSTANCIA

Vista la tesis denominada: "IMPLEMENTACIÓN DE UNA APLICACIÓN MÓVIL COMPLEMENTADA CON LA TECNOLOGÍA DE SENSORES MULTIPARAMÉTRICOS CON ARDUINO PARA MONITOREAR LA CALIDAD DEL AGUA POTABLE EN EL DISTRITO DE LAREDO, TRUJILLO", perteneciente a los Bach. Raúl Alfonso Ocola Gadea y Bach. Gorky Walter Ramos Mendoza, se deja constancia que los resultados obtenidos producto de su investigación, en contraste con los resultados de los análisis realizados por parte del Área Técnica Municipal (ATM) del Distrito de Laredo, a cargo del Ing. Juan Manuel Liñán Rodríguez, son válidos, por lo que se recomienda su aplicación.

Se refrenda la presente para fines que los autores crean conveniente.

Trujillo, 25 de octubre de 2018

 MUNICIPALIDAD DISTRITAL
DE LAREDO

Lic. Georiana V. Guzman Rivas
GERENTE MUNICIPAL.