AAT6358

UCAB Universidad Católica ANDRÉS BELLO

FACULTAD DE INGENIERÍA

ESCUELA DE INGENIERÍA INFORMÁTICA

SISTEMA DE GESTIÓN PARA PROCESOS MEDULARES EN EL ESTUDIO DE MUESTRAS PARA LA UNIDAD DIAGNÓSTICO MOLECULAR DE LA FUNDACIÓN JACINTO CONVIT

TRABAJO INSTRUMENTAL DE GRADO

Presentado ante la

UNIVERSIDAD CATÓLICA ANDRÉS BELLO

Como parte de los requisitos para optar al título de

INGENIERO EN INFORMÁTICA

REALIZADO POR	**MARIANGELA DELGADO**
TUTOR EMPRESARIAL	**YANELIZ LAFEE**
TUTOR ACADÉMICO	**GLORIA TARRÍO**
FECHA	**FEBRERO 2018**

UCAB Universidad Católica ANDRÉS BELLO

FACULTAD DE INGENIERÍA

ESCUELA DE INGENIERÍA INFORMÁTICA

SISTEMA DE GESTIÓN PARA PROCESOS MEDULARES EN EL ESTUDIO DE MUESTRAS PARA LA UNIDAD DIAGNÓSTICO MOLECULAR DE LA FUNDACIÓN JACINTO CONVIT

Este Jurado; una vez realizado el examen del presente trabajo ha evaluado su contenido con el resultado: Diecinueve (19)

JURADO EXAMINADOR

Firma:	Firma:	Firma:
Nombre: Carlo Magnano	Nombre: Carlos Alonso	Nombre: Gloria Tarrío V.

REALIZADO POR	**MARIANGELA DELGADO**
TUTOR EMPRESARIAL	**YANELIZ LAFEE**
TUTOR ACADÉMICO	**GLORIA TARRÍO**
FECHA	**FEBRERO 2018**

DEDICATORIA

Quiero dedicar este Trabajo Instrumental de Grado primero que todo a Dios, por regalarme cada día para trabajar y darme la fuerza y salud necesarias para continuar.

A mi mamá por darme todo lo que ha podido, su tiempo, esfuerzo y amor son invaluables y quiero dedicarle cada logro que alcance durante mi vida.

A mi madrina Rosario Meza quien creyó siempre en mí a pesar de mis tropiezos y aunque no pudiste estar en este momento, sé que desde donde estés compartes mi alegría.

Al Sr. Enrique Bucherenick quién confío en mí desde que inicié este trabajo y con su cariño, humor y consejos me contagiaba de ánimo para seguir porque debía culminar y él sabía que podría lograrlo.

AGRADECIMIENTOS

A mi mamá por todo su amor, su apoyo y confianza desde el inicio de mi carrera, por estar a mi lado en mis malos momentos y regalarme sus fuertes abrazos. A mi papá por todo su apoyo y por continuar creyendo en mí.

A la Sra. Dulce y al Sr Enrique, por abrirme las puertas de su hogar y hacerme sentir como una hija, por su apoyo y cariño continuo.

A Marco Franceschi, por impulsarme a continuar con este trabajo, por confiar y ver en mí lo que a veces yo misma no logro ver, por tu tiempo, paciencia y consejos. También quiero agradecer a Eilin y Andrea por su valiosa ayuda.

A mis amigas Andreina, Roxana, Yolymar, Viviana y Joali por acompañarme durante la carrera, por su cariño incondicional y por todo su apoyo que aún en la distancia siempre me hicieron sentirlas cercanas.

A Mimia Lo por sus consejos y apoyo durante la realización de este trabajo y a todos los que de alguna u otra forma hicieron parte de este trabajo y mi carrera.

A mi tutora Gloria Tarrío, gracias por su apoyo y ayuda durante la realización de este trabajo.

A la Sra. Berti, al Sr. Rolando, al Sr. Hector, a la Sra. Rosaura, Teresa y a la Sra. Lucy por tanto cariño, por su confianza, su apoyo y todo ese ánimo que me han dado para culminar esta etapa tan importante.

A mi familia, gracias, por su amor y su apoyo aún en la distancia.

A la Fundación Jacinto Convit por confiar en mí y darme la oportunidad de desarrollar este sistema tan importante para ustedes.

Finalmente quiero agradecer a la Fundación Andrés Bello por el apoyo económico que recibí durante mi carrera y por supuesto a la UCAB y a todos mis profesores por toda la formación que obtuve como profesional.

TABLA DE CONTENIDOS

Dedicatoria i
Agradecimientos ii
Índice de figuras vi
Índice de tablas vii
Sinopsis viii
Introducción ix
CAPÍTULO I. Planteamiento del problema 1
I.1 Necesidad de la empresa 1
I.2 Solución Propuesta 4
I.3 Objetivos 7
I.3.1 Objetivo General 7
I.3.1 Objetivos Específicos 7
I.4 Aportes 8
I.4.1 Aportes Tecnológicos 8
I.4.2 Aporte Funcional 8
I.5 Alcance y Limitaciones 8
I.5.1 Alcance 8
I.5.2 Limitaciones 15
I.6 Justificación 15
CAPÍTULO II – Marco Referencial 17
II.1. Análisis 17
II.1.1. Definición 17
II.1.1. PCR 17
II.1.2. Reactivo 18
II.2 Sistema Web 18
II.2.1 Definición 18
II.2.2 Servidor Web 18
II.3. JAVA 19
II.4. Spring 19

II.4.1 Inyección de dependencia 20
II.5. Spring Boot 20
II.5.1 Características 20
II.6 Spring Data JPA 20
II.7. HTML5 21
II.8. CSS 21
II.9 Bootstrap 21
II.10. Angular 22
II.11. TomCat 22
II.12. PostgreSql 22
II.13. Servicios REST 22
II.14 JWT (JSON Web Tokens) 23
CAPÍTULO III – Marco metodológico 24
III.1. Crystal 24
II.2. Crystal Clear 26
III.2.1. Roles de Crystal Clear 26
II.2.2. Valores, propiedades y técnicas 27
III.2.3. Justificación de selección 29
III.2.4. Proceso 30
CAPÍTULO IV – Desarrollo 33
IV.1. Fase de exploración 33
IV.1.1 Evaluaciones 36
IV.1.2 Arquitectura 42
IV.1.3 Usuarios 45
IV.2. Incremento 1 46
IV.2.1. Iteraciones 46
IV.3. Incremento 2 49
IV.3.1 Iteraciones 49
IV.4. Incremento 3 52
IV.4.1. Iteraciones 52

IV.5. Incremento 4 56
IV.5.1 Iteraciones 56
IV.6. Incremento 5 58
IV.6.1. Iteraciones 58
CAPÍTULO V - Resultados 60
CAPÍTULO VI. Conclusiones y recomendaciones 65
VI.1 Conclusiones 65
VI.2 Recomendaciones 67
Referencias Bibliográficas 68
APÉNDICES 70
Apéndice A - Glosario 70
Apéndice B – Exámenes ofrecidos por la UDM 72
Apéndice C - Requerimientos funcionales e historias de usuario 73
Apéndice D – Encuestas y evaluación de tecnologías 79
Apéndice E – Modelo Entidad - Relación 81
Apéndice F- Ilustraciones del sistema 82
Apéndice G- Esquema de condiciones para la PCR 90
Apéndice F- Fórmula para el cálculo del volumen 91

ÍNDICE DE FIGURAS

Figura 1 Pérdidas que puede ocasionar la falla de un sistema 25
Figura 2 Proceso de Crystal Clear adaptado al proyecto 31
Figura 3 Interacción del sistema 34
Figura 4 Configuración inicial para crear una aplicación con JHipster 38
Figura 5 Diagrama del proyecto creado con JDL Studio 39
Figura 6 Arquitectura del framework Angular 44
Figura 7 Arquitectura del sistema 45
Figura 8 Menú de opciones del sistema 48
Figura 9: Tecnologías más populares en el 2016 según Stack Overflow 79
Figura 10: Otras tecnologías más popularmente deseadas del 2016 según Stack Overflow 79
Figura 11: Top de los lenguajes de programación según TIOBE 80
Figura 12: Formulario del registro de usuario 82
Figura 13: Listado de usuarios del sistema 82
Figura 14: Pantalla de edición de usuario 82
Figura 15: Inicio de sesión del sistema 83
Figura 16 : Cambio de contraseña 83
Figura 17: Recuperación de contraseña 83
Figura 18: Listado de casas comercial 83
Figura 19: Listado de unidades de medida 84
Figura 20: Pantalla de inventario de productos 84
Figura 21: Creación de un producto 84
Figura 22: Creación de un reactivo 85
Figura 23: Creación de pruebas en la wiki 85
Figura 24: Cálculos de una prueba. Módulo de cálculos 86
Figura 25: Fase de pre-análisis 86
Figura 26: Selección de muestras. Fase de análisis 87
Figura 27: Fase de análisis 87
Figura 28: Fase de post-análisis 88
Figura 29: Configuración de notificaciones 88
Figura 30: Histórico de muestras 88
Figura 31: Detalle de la información de la muestra 89
Figura 32: Módulo de estadísticas 89
Figura 33 Master mix de una PCR 90
Figura 34 Programación para PCR 90
Figura 35: Ejemplo del código aplicado al cálculo del volumen de un reactivo. 92

ÍNDICE DE TABLAS

Tabla 1 Roles de la metodología designados en el proyecto 29
Tabla 2 Fases y actividades del proyecto 32
Tabla 3 Historia de usuario 001 75
Tabla 4 Historia de usuario 002 75
Tabla 5 Historia de usuario 003 75
Tabla 6 Historia de usuario 004 75
Tabla 7 Historia de usuario 005 75
Tabla 8 Historia de usuario 006 75
Tabla 9 Historia de usuario 007 76
Tabla 10 Historia de usuario 008 76
Tabla 11 Historia de usuario 009 76
Tabla 12 Historia de usuario 010 76
Tabla 13 Historia de usuario 011 76
Tabla 14 Historia de usuario 012 76
Tabla 15 Historia de usuario 013 77
Tabla 16 Historia de usuario 014 77
Tabla 17 Historia de usuario 015 77
Tabla 18 Historia de usuario 016 77
Tabla 19 Historia de usuario 017 77
Tabla 20 Historia de usuario 018 78
Tabla 21 Historia de usuario 019 78
Tabla 22 Historia de usuario 020 78
Tabla 23 Historia de usuario 021 78

SINOPSIS

El presente Trabajo Instrumental de Grado tiene por título "Sistema de gestión para procesos medulares en el estudio de muestras para la Unidad Diagnóstico Molecular de la Fundación Jacinto Convit", se basa en la creación de un sistema para optimizar el proceso de ejecución de pruebas de bionálisis a las muestras que ingresan a la Fundación Jacinto Convit, así como también la gestión de un inventario para obtener información actualizada de los insumos utilizados cada día.

La Unidad Diagnóstico Molecular (UDM) tiene la visión de aportar mejoras en el sistema de diagnóstico, control, prevención y tratamiento de enfermedades. Aunque la unidad ejecuta cada día pruebas a las distintas muestras que ingresan a la Fundación, sus procesos no se encuentran centralizados y existe una inversión de tiempo y esfuerzo de parte del equipo del laboratorio para la aplicación de los análisis. Por esta razón se propone una solución que divide la construcción en dos módulos principales, uno de ellos es un inventario de insumos y el segundo es el proceso encargado de la ejecución de pruebas, éste además estará en comunicación con el módulo de inventario.

Este Trabajo Instrumental de Grado se desarrolló bajo el marco de trabajo Crystal Clear, el cual presenta un enfoque ágil, haciendo gran énfasis en la comunicación con el cliente manejando iteraciones cortas con *feedback* frecuente y de forma muy liviana en relación a los documentos entregables.

INTRODUCCIÓN

La salud, es considerada el pilar para realizar cualquier actividad que una persona anhele concebir, es por esto que ante la prevención y detección temprana de enfermedades cada día se realizan diferentes análisis para diagnosticar el estado de salud de una persona. La Fundación Jacinto Convit dedica sus esfuerzos en generar investigación científica y programas asistenciales de salud, para contribuir con el bienestar de las poblaciones más vulnerables y con el desarrollo social del país, por estas razones posee la Unidad de Diagnóstico Molecular (UDM), que tiene dentro de sus objetivos elevar los niveles de diagnóstico, atención y tratamiento de los pacientes con enfermedades crónicas e infecciosas mediante la investigación básica y clínica.

Es de vital importancia llevar un proceso centralizando en la búsqueda de optimizar los procesos de la Fundación. El presente Trabajo Instrumental de Grado describe el desarrollo de un sistema web para la gestión y control de procesos medulares de la Unidad de Diagnóstico Molecular de la Fundación Jacinto Convit.

A continuación se describe el contenido de los seis capítulos en los que se dividió este Trabajo Instrumental de Grado:

- Capitulo I. Planteamiento del problema

En este capítulo, se describe en detalle la problemática que tiene la Fundación Jacinto Convit, así como también la solución planteada para su resolución. Se dividió en objetivos, alcance, limitaciones y justificación.

- Capitulo II. Marco Teórico

Este capítulo es el que sustenta los conceptos y definiciones presentados en este Trabajo Instrumental de Grado, explicando los términos del negocio, así como las tecnologías y herramientas utilizadas durante el desarrollo.

- Capítulo III. Marco Metodológico

En este capítulo se describe el marco metodológico empleado durante el desarrollo del sistema, detallando los roles, valores y el proceso que se emplea al aplicar la metodología.

- Capitulo IV. Desarrollo

Se explica detalladamente la arquitectura aplicada, el grupo de usuarios que accede a la aplicación, así como también cada una de las entregas realizadas en el desarrollo del sistema y su funcionamiento.

- Capítulo V. Resultados

Este capítulo describe el producto obtenido, explicado por cada uno de los objetivos específicos planteados.

- Capítulo VI. Conclusiones y Recomendaciones

En este capítulo se describen las conclusiones y aprendizajes obtenidos al finalizar el desarrollo, así como también las recomendaciones o mejoras que puedan realizarse para ampliar el proyecto.

CAPÍTULO I. PLANTEAMIENTO DEL PROBLEMA

I.1 Necesidad de la empresa

Desde siempre la salud ha sido un tema de suma importancia para cualquier persona y la base para desarrollar una vida que no sólo sea prolongada, sino plena; los avances científicos y tecnológicos en los tiempos modernos han permitido que la expectativa de vida de las personas se incremente de manera dramática. Una de las principales formas es a través de la prevención y detección temprana de enfermedades.

Uno de los problemas centrales a los que se enfrenta una parte de la población es el diagnóstico tardío y no preciso de enfermedades así como el acceso limitado a los sistemas de control y tratamientos.

La Fundación Jacinto Convit tiene como misión contribuir con el desarrollo social a través de la investigación científica, programas asistenciales de salud, proyectos educativos y comunitarios que incidan en el bienestar de las poblaciones más vulnerables. La Fundación cuenta con proyectos como la Biblioteca Digital Jacinto Convit y Valores a través de la ciencia en el área educativa, en el área social con el Proyecto Waraos y en el área de Investigación con la Unidad Experimental de Inmunoterapia (UEI) y la Unidad de Diagnóstico Molecular (UDM).

La UDM tiene como objetivo elevar los niveles de diagnóstico, atención y tratamiento de los pacientes con enfermedades crónicas e infecciosas mediante la investigación básica y clínica, basándose en las técnicas

moleculares avanzadas que permitan brindar un servicio de calidad y gratuito. En el Apéndice B se encuentra una lista con los exámenes ofrecidos por la UDM.

Cada día ingresan a los laboratorios de la Fundación, una cantidad de muestras considerables que deben ser analizadas generando sus respectivos diagnósticos. El diagnóstico de muestras comprende tres fases asociadas: la fase de Pre-análisis, muestras que ingresan a la UDM de la Fundación; el Análisis, en el cual se analiza el material genético aislado de las muestras de acuerdo a la solicitud del examen; y finalmente, la fase de Post-análisis donde se verifican los resultados obtenidos en el análisis para emitir un diagnóstico molecular para cáncer infantil y enfermedades infecciosas.

Por cada muestra recibida en la UDM para realizar el diagnóstico molecular, los investigadores deben realizar el siguiente procedimiento: realizar el registro de la muestra ingresada en archivos; posteriormente, deben identificar el procedimiento adecuado a seguir con dicha muestra, consultando fuentes de conocimiento diversas y no centralizadas, las cuales, en su mayoría está contenida en medios físicos como cuadernos, ficheros, carpetas entre otros. Una vez determinado el procedimiento a seguir, deben identificar los distintos materiales, instrumentos y reactivos necesarios, determinar su ubicación, registrar en cuadernos las distintas cantidades de cada reactivo a usar, así como también llevar a cabo cálculos manuales usando distintas

fórmulas (obtenidas de las fuentes de conocimiento mencionadas anteriormente) que permiten calcular la composición correcta.

Todo el proceso descrito anteriormente se hace de forma manual, lo que consume mucho tiempo y esfuerzo, además de hacerlo más vulnerable al error humano; también es importante destacar que el proceso no está formalmente definido ni sigue una estructura que permita realizar un monitoreo o seguimiento histórico del mismo.

La UDM no cuenta con un inventario automatizado de reactivos y consumibles, que permita llevar el control de los mismos y que facilite el trabajo de los operadores de los laboratorios. Los procesos de consulta no se encuentran automatizados dado que deben verificar frecuentemente la existencia de algún componente dentro del inventario (inventario digital como en un documento tipo Excel o su búsqueda física en los almacenes); inversión de tiempo y trabajo que puede ser reducido y facilitado con la centralización de la información. Además, no se cuenta con un mecanismo que alerte sobre el reintegro o reposición de componentes cuando estos están próximos a terminarse, sin que tenga que afectar la productividad del personal médico de la Fundación.

Es de gran importancia para la Fundación Jacinto Convit contar con recursos que los apoyen en procesos medulares, que ayuden mejorar y en algunos casos cambiar o rediseñar los mismos, en vías de minimizar errores, disminuir tiempo y facilitar el trabajo que desarrollan, incrementando la

confiabilidad, sirviendo como una base de implementación de estándares de calidad, dada la importancia de elevar los niveles de diagnóstico, atención y tratamiento de los pacientes con enfermedades crónicas e infecciosas mediante la investigación básica y clínica.

I.2 Solución Propuesta

Para llevar a cabo la gestión y control de las distintas fases que componen el proceso de análisis de una muestra, así como también la información de los procedimientos para dichos estudios, cálculo de cantidades y proporciones para lograr una composición correcta, uso de los distintos reactivos y consumibles, entre otros, descritos de manera sucinta en la presentación del problema; se propone, el desarrollo de un sistema web de acceso exclusivo para personal autorizado desde la intranet de la Fundación; que no sólo permita la organización, automatización, consolidación y control de los procesos medulares para la Fundación, sino también el mejoramiento de los mismos y los resultados derivados, por medio de su rediseño, apoyándose en las nuevas posibilidades que brindará un sistema de estas características por sobre las condiciones actuales.

El ingreso al sistema web se realizará a través de credenciales de usuarios y contará con un mecanismo de roles, todo ello gestionado desde una sección de administración de usuarios.

Se podrán registrar los datos de las muestras y sus solicitudes de análisis, el sistema les asignará un identificador único para realizar su seguimiento. Se contará con una sección, al estilo *wiki* (espacio de escritura editable), donde estarán definidos los pasos a seguir para realizar las diferentes pruebas, cantidad de reactivos que debe utilizarse e insumos necesarios, además que para cada reactivo se mostrará su valor final (concentración y volumen necesario), dejando así esta información de gran importancia como una fuente de conocimiento centralizada.

Para lograr la composición correcta al aplicar una prueba en el laboratorio se requiere de cálculos matemáticos, el sistema realizará los cálculos necesarios para aplicar un tipo de prueba al recibir los parámetros iniciales (para algunas pruebas: volumen de la muestra o cantidad de muestras). Los datos obtenidos serán guardados y asociados a la muestra, posteriormente el personal de la Fundación podrá registrar los resultados obtenidos al finalizar el examen diagnóstico.

En cada fase de una muestra podría aplicarse más de un examen diagnóstico, por esta razón se tendrá la opción de identificar a qué fase de análisis pertenece una prueba. De cada muestra podrá llevarse un registro o histórico que refleje las fases, las pruebas por las que ha pasado, los usuarios del laboratorio que han realizado estudios sobre la muestra y el diagnóstico final.

El sistema también contará con un inventario automatizado que contendrá todos los reactivos y consumibles que estén almacenados en la Fundación, de este inventario se decrementará de forma automática la cantidad de reactivos que se hayan utilizado en cada una de las pruebas diagnósticas e investigación. Se establecerá una sección de notificaciones para informar al personal de la Fundación cuando la cantidad de un producto esté por llegar al límite en el que se debe reabastecer del mismo, este límite será establecido por un administrador dentro del sistema.

Finalmente el sistema contendrá una sección de estadísticas en la que se podrá consultar distintos datos relevantes como la cantidad de insumos que se están consumiendo en pruebas, cuantas muestras pueden ser analizadas con la cantidad total de un envase de reactivo específico, entre otros.

I.3 Objetivos

I.3.1 Objetivo General

Desarrollar un sistema *web* para la gestión y control de procesos medulares de la Unidad de Diagnóstico Molecular de Fundación Jacinto Convit.

I.3.1 Objetivos Específicos

1. Diseñar e implementar un modelo de base de datos que se adapte a la solución propuesta.
2. Diseñar e implementar un módulo de roles, permisos y gestión de usuarios.
3. Diseñar e implementar un módulo de pruebas y análisis que aloje información sobre su implementación.
4. Diseñar e implementar un módulo inventario de insumos, reactivos y consumibles.
5. Diseñar e implementar un módulo de cálculos y fórmulas matemáticas.
6. Diseñar e implementar un módulo de registro de procesos de muestras.
7. Diseñar e implementar un módulo de alertas de fallos en el almacenamiento.
8. Diseñar e implementar un módulo histórico de análisis.
9. Diseñar e implementar un Módulo de estadísticas.

I.4 Aportes

I.4.1 Aportes Tecnológicos

1. Evaluar la plataforma de desarrollo y seleccionar la más adecuada para la programación de los módulos propuestos para el sistema web.
2. Integrar el sistema web con el sistema de gestión de mantenimiento.

I.4.2 Aporte Funcional

Rediseñar los procesos medulares actuales aplicados al análisis de muestras que ingresan a la Fundación Jacinto Convit.

I.5 Alcance y Limitaciones

I.5.1 Alcance

A continuación, se presenta una descripción detallada de los objetivos para la realización del sistema web:

1. **Diseñar e implementar un modelo de base de datos que se adapte a la solución propuesta.**

Dada la necesidad de almacenar la data de las pruebas que se realizan a la muestra, se creará una base de datos tomando en cuenta la alta granularidad en el diseño para poder realizar seguimiento de muestras, información detallada de los análisis, llevar el almacenamiento de los insumos y consumibles, almacenar los usuarios registrados en el sistema y demás data

relacionada que requiera el sistema web para su manejo y persistencia de datos.

2. **Diseñar e implementar un módulo de roles, permisos y gestión de usuarios.**

Este módulo permitirá a los administradores crear usuarios con diferentes permisologías, de esta forma se otorgará acceso a la edición del inventario o edición del módulo de conocimiento a ciertos usuarios.

3. **Diseñar e implementar un módulo de prueba y análisis que aloje información sobre su implementación.**

Este módulo permitirá gestionar información referente a las pruebas y análisis. Se clasificarán las pruebas que pueden realizarse en los laboratorios de la Fundación, para cada una se mostrará el procedimiento para su implementación especificando las tareas, pasos, técnicas, componentes y reactivos que deben usarse así como la cantidad de estos, además para ciertos reactivos se mostrará su valor final (concentración y volumen necesario). A través de este módulo se facilitará el manejo de la información al estar centralizada, además de ser un apoyo para quienes se incorporen como nuevo personal al laboratorio. Los usuarios con permiso de edición podrán gestionar toda la información de este módulo.

4. **Diseñar e implementar un módulo inventario de insumos, reactivos y consumibles.**

Se diseñará e implementará un módulo de administración del almacenamiento de los recursos usados diariamente en la UDM. Este módulo estará enlazado a la creación de pruebas, al aplicar un estudio a una muestra, el sistema restará automáticamente los componentes usados en la prueba de la cantidad almacenada. Este módulo de inventario también deberá almacenar la ubicación dentro del laboratorio de cada ítem registrado.

5. **Diseñar e implementar un módulo de cálculos y fórmulas matemáticas.**

Este módulo permitirá automatizar los cálculos de fórmulas metamatemáticas, necesarias para la realización de diferentes análisis, permitirá el ingreso de parámetros según la prueba que se ejecute. Las diferentes pruebas que se realizan en la Fundación deberán estar cargadas en el módulo de información de pruebas y análisis.

6. **Diseñar e implementar un módulo para el registro de procesos de muestras.**

Este módulo permitirá la creación de las muestras dentro del sistema, registrando sus datos y la orden médica que detalla las pruebas que deban realizarse. A cada muestra se le asignará un identificador único. Además será a través de este módulo, que el personal del laboratorio podrá agregar las

pruebas que se estén realizando a la muestra y el resultado que se obtenga en las pruebas al aplicarlas.

7. **Diseñar e implementar un módulo de alertas de fallos en el almacenamiento.**

Se diseñará e implementará un módulo para el envío de notificaciones cuando algún recurso esté por llegar al límite. Este módulo enviará alertas a través de correos electrónicos para los usuarios que seleccione el administrador del sistema.

8. **Diseñar e implementar un módulo histórico de análisis.**

Este módulo permitirá llevar un monitoreo de la muestra en cada una de sus fases, así como consultar los diagnósticos presuntivos y finales en la muestra.

9. **Diseñar y desarrollar un módulo de estadísticas**

Permitirá hacer un seguimiento acertado de la cantidad de insumos que está utilizando el personal de la Fundación, además, realizar consultas que permitan saber:

- Cuántas pruebas pueden realizarse con todo el contenido de un componente o reactivo.
- Cuántas muestras puede cubrir un reactivo o una encima.

Alcance de los aportes tecnológicos:

1. **Evaluar la plataforma de desarrollo y seleccionar la más adecuada para la programación de los módulos propuestos para el sistema web.**

Se realizará la evaluación de distintas plataformas de desarrollo para determinar cuál es la más conveniente para desarrollar el sistema, esto tomando en cuenta principalmente los recursos con los que cuenta la organización, así como también diferentes aspectos como mantenibilidad, escalabilidad, portabilidad, entre otros.

Se seleccionará un lenguaje de programación orientado a la resolución de problemas matemáticos, para el desarrollo del módulo de cálculos y fórmulas matemáticas, esto buscando la optimización de las operaciones. Se integrará este lenguaje al seleccionado para el desarrollo del sistema web.

2. **Integrar el sistema web con el sistema de gestión de mantenimiento.**

Se llevará a cabo la integración del sistema web a desarrollar con el sistema de gestión de mantenimiento que ya posee actualmente la Fundación. Este sistema almacena toda la información de uso de cada equipo científico de los laboratorios de la Fundación, incluyendo los detalles de su desempeño.

El objetivo de integrar ambos sistemas es tener acceso desde el inventario a los equipos de la Fundación, además de los insumos, reactivos y

consumibles. Con esta integración se busca evitar la duplicidad de información.

Alcance del aporte funcional:

1. **Rediseñar los procesos medulares actuales aplicados al análisis de muestras.**

En la actualidad diariamente se reciben en los laboratorios de la Fundación, numerosas muestras a las que se le realizan diversas pruebas para lograr un diagnóstico final. Esto involucra varios procesos relacionados que no están claramente definidos ni detallados, estos procesos son, el proceso de análisis; el proceso de control y mantenimiento del inventario; y el proceso de control y seguimiento de usuarios, muestras y pruebas.

En cuanto al proceso de análisis se tiene que, no se cuenta con un proceso estructurado para realizar el registro de las muestras que ingresan y las pruebas que se le realizan a las mismas; la información sobre los procedimientos a implementar se encuentra en diversos medios físicos dispersos, no se tiene una forma de obtener rápidamente la información requerida de los mismos ni la ubicación de los insumos necesarios; los cálculos para obtener las composiciones correctas y poder aplicar una prueba se realizan de forma manual, así como el registro de los insumos usados e información adicional de la prueba, no existe una forma de evitar posibles errores humanos en los cálculos y registros. El control del inventario con el que se trabaja es solo un documento Excel el cual debe ser actualizado y

consultado constantemente, o dirigirse al almacén para saber cuánto queda de un reactivo para reponerlo. No se cuenta con un mecanismo específico para el control y seguimiento de usuarios, muestras y pruebas; actualmente se realiza a través de anotaciones en cuadernos y documentos Excel, el proceso dependerá mucho de la persona que realice la prueba, por lo que ubicar datos históricos, realizar seguimiento, relacionar variables, etc., se presenta particularmente difícil.

Con el rediseño de estos procesos se busca suprimir pasos innecesarios que aumentan el tiempo, la complejidad de las tareas, sus costos y posibles errores humanos, así como también, se busca poder llevar a cabo el monitoreo y seguimiento de cada una de las etapas que componen los mismos, llevar un control de sus entradas-salidas y poder contar con un registro histórico.

El rediseño que se llevará a cabo se describe brevemente a continuación: se eliminaran pasos innecesarios como la recolección y registro manual en archivos de los datos de las pruebas, la búsqueda de información por parte del personal de la Fundación en distintas fuentes dispersas sobre los procedimientos para realizar una prueba determinada a una muestra, la búsqueda de los insumos necesarios para realizar una prueba y su ubicación física, los cálculos manuales para la obtención de composiciones correctas para realizar un análisis, el registro por parte del personal de la Fundación de las distintas cantidades de componentes e insumos utilizados en una prueba; se modificará el mecanismo mediante el cual se controla el inventario

realizando actualizaciones automáticas del mismo, adicionalmente se añadirán alarmas de reposición de insumos con las que no se cuentan en la actualidad; se añadirá la posibilidad de llevar un control histórico de las distintas muestras, pruebas realizadas, insumos usados en cada prueba, así como también, la relación de cada prueba con el usuario encargado de realizarla y su conexión con el inventario de insumos.

I.5.2 Limitaciones

- El ingreso al sistema será exclusivamente para el personal de la Fundación Jacinto Convit.
- El manejador de base de datos que se usará es *PostgreSql*.

I.6 Justificación

Actualmente la Fundación Jacinto Convit cuenta con la UDM, con la cual aportan mejoras en el sistema diagnóstico, control y prevención de enfermedades con el fin de contribuir en la obtención de un mejor sistema de salud. Sus procesos medulares los conforman la investigación, diseño y desarrollo de protocolos de diagnóstico molecular por Reacción en cadena de la polimerasa (PCR por sus siglas en inglés) para la identificación de anormalidades cromosómicas y mutaciones asociadas a leucemias y tumores sólidos cerebrales. Para cada una de las fases que conforman el análisis de muestras (pre-análisis, análisis y post-análisis) no existe una herramienta

centralizada que permita gestionar cada etapa, almacenar los mecanismos utilizados en la aplicación de análisis o conservar un histórico de las pruebas realizadas.

El control de inventario de consumibles y reactivos es también un problema dentro de la Fundación, actualmente para conocer la existencia de algún producto se debe consultar un documento Excel o dirigirse al almacén. De allí nace la necesidad de crear un inventario automatizado que además de contener los insumos de la Fundación también disminuya de forma automática los reactivos aplicados en un análisis.

Por lo anteriormente descrito se plantea el siguiente Trabajo Instrumental de Grado con el objetivo de desarrollar un sistema de gestión y control de los procesos que permitan dar respuesta a la problemática antes mencionada.

CAPÍTULO II – MARCO REFERENCIAL

II.1. Análisis

II.1.1. Definición

Un análisis es el acto de separar las partes de un elemento para estudiar su naturaleza, su función y/o su significado. Un análisis es un efecto que comprende diversos tipos de acciones con distintas características y en diferentes ámbitos, pero resumidamente es todo acto que se realiza con el propósito de estudiar, ponderar, valorar y concluir respecto de un objeto, persona o condición [1]. Entre los principales análisis que se ejecutan en la Fundación Jacinto Convit se encuentran las pruebas *Polymerase chain reaction* (PCR).

II.1.1. PCR

Polymerase chain reaction (PCR) o Reacción en cadena de la polimerasa es una manera eficiente y rentable de copiar o "amplificar" pequeños segmentos de ácido desoxirribonucleico (ADN) o ácido ribonucleico (ARN). Usando PCR, se pueden hacer millones de copias de una sección de ADN, obteniendo suficiente ADN necesario para el análisis. Este método innovador permite a los médicos diagnosticar y monitorear las enfermedades usando una cantidad mínima de muestra como sangre o tejido. En su forma más simple, la reacción de replicación del ADN se produce cuando se añade una muestra de ADN y un ADN polimerasa, nucleótidos, cebadores y otros reactivos a un tubo de muestra [4].

II.1.2. Reactivo

Un reactivo o reactante es toda sustancia que interactúa con otra en una reacción química o biológica que da lugar a otras sustancias de propiedades, características y conformación distinta, denominadas productos de reacción o simplemente productos. Por tratarse de compuestos químicos, los reactivos se pueden clasificar según muchas variables: propiedades fisicoquímicas, reactividad en reacciones químicas, características del uso del reactivo.

II.2 Sistema Web

II.2.1 Definición

Los sistemas web se definen como aquellas aplicaciones a las cuales puede accederse a través de un navegador web, quien, por el ingreso de un localizador de recurso uniforme (URL, por sus siglas en inglés) hace una petición a un servidor web a través de una intranet o Internet. A diferencia de una página web, los sistemas web buscan interactuar con el usuario, aportando más que una fuente de información [14].

II.2.2 Servidor Web

Un servidor web es un programa de *software* desarrollado para gestionar peticiones del lado del servidor. Se basa en realizar conexiones bidireccionales y/o unidireccionales con el cliente web, comunicándose bajo el protocolo de transferencia de hipertexto (HTTP, por sus siglas en inglés) quien permite la trasferencia de información o respuestas generadas, el código que

recibe el cliente como respuesta es renderizado por un navegador web, quien se encarga de visualizar las páginas web [14].

II.3. JAVA

Java es un lenguaje de programación basado en el concepto de programación orientada a objetos (POO). Es un lenguaje derivado de C, por lo que sus reglas y sintaxis se parecen mucho a C [13]. Es definido como un lenguaje interpretado y compilado a la vez, dado que se compila en la medida que su código fuente se transforma a código de máquina e interpreta porque los *bytecodes* se pueden ejecutar directamente sobre cualquier máquina, ya que usa una máquina virtual de java (JVM, por sus siglas en inglés), que es donde realmente se ejecuta el código y no sobre el sistema operativo. Java es también considerado robusto, de alto rendimiento y portable.

II.4. Spring

Spring es un *Framework* que proporciona un modelo de programación y configuración completa para las aplicaciones basadas en Java en cualquier tipo de plataforma de despliegue. Con algunas características como la inyección de dependencia y simplificación al integrarse con *Java Persistence API* (JPA). Spring proporciona flexibilidad para la configuración de aplicaciones, como *XML* y anotaciones, sin embargo, al aumentar las características, la complejidad también se incrementa y la configuración de aplicaciones se vuelve tediosa y propensa a errores [6].

El equipo de Spring creó SpringBoot para abordar la complejidad de la configuración.

II.4.1 Inyección de dependencia

Es una herramienta comúnmente utilizada en varios patrones de diseño orientado a objetos, consiste en inyectar comportamientos a componentes [15].

II.5. Spring Boot

Es una tecnología dentro del mundo de Spring orientada a facilitar la creación de aplicaciones independientes, de calidad y producción con Spring *Framework*, proporcionando configuraciones por defecto para este y gran cantidad de librerías [16].

II.5.1 Características

- Provee una forma sencilla de ejecutar el código desarrollado, con el servidor de aplicaciones embebido *Tomcat*, *Jetty* o *Undertow*, directamente sin necesidad de desplegar archivos de aplicación web (*WAR*, por sus siglas en inglés).
- Spring configura automáticamente siempre que sea posible.
- No genera código y no requiere configuración de XML [16].

II.6 Spring Data JPA

Forma parte de la familia de librerías de Spring Data pertenecientes al *framework* Spring, su objetivo es mejorar significativamente el manejo de la

capa de acceso a datos, poniendo en práctica repositorios (*Repository*) [17]. Se le puede considerar como una capa abstracta montada sobre JPA, de esta forma provee de varias interfaces heredables que contienen métodos necesarios para el acceso a base de datos.

II.7. HTML5

HTML es el lenguaje usado para el desarrollo de páginas web, con él se define la estructura del contenido de las páginas, su última versión, HTML5 contiene nuevos elementos atributos y comportamientos, planteado para ser usables por todos los desarrolladores de la web [11].

II.8. CSS

Cacading Style Sheets u Hoja de estilo en cascada (CSS), es un lenguaje utilizado para describir la presentación de documentos HTML o XML. CSS describe como debe ser renderizado el elemento estructurado en pantalla [10].

II.9 Bootstrap

Es un *framework Open Source* o código abierto creado por Twitter Inc. Con el objetivo de simplificar la creación de interfaces web, al ofrecer una variedad de opciones y estilos para el diseño de estas, que pueden implementarse de forma rápida, basándose en HTML y CSS y haciendo uso de *JavaScript*.

II.10. Angular

Es un *framework* creado por el equipo de Google, para la construcción de aplicaciones web, usando HTML y lenguaje *TypeScript* (que se compila en *JavaScript*), combina plantillas declarativas, inyección de dependencia, herramientas de extremo a extremo para ser aplicadas en el desarrollo *FrontEnd* [2].

II.11. TomCat

Desarrollado por miembros de *Apache Software Foundation* y escrito en Java, Tomcat es una solución de código abierto que funciona como un contenedor de *servlets,* el cual es un programa capaz de recibir peticiones de páginas web y redireccionar estas peticiones a un objeto *Servlet* [5].

II.12. PostgreSql

Es un sistema de gestión de bases de datos objeto-relacional, distribuido bajo licencia BSD (*Berkeley Software Distribution*) y con su código fuente disponible libremente. Utiliza un modelo cliente/servidor y usa multiprocesos en vez de multihilos para garantizar la estabilidad del sistema. Un fallo en uno de los procesos no afectará el resto y el sistema continuará funcionando [9].

II.13. Servicios REST

La Transferencia de Estado Representacional (REST, por sus siglas en inglés), fue definido en el año dos mil (2000) por Roy Fielding en su tesis doctoral. Es un estilo de arquitectura utilizado para la implementación de

aplicaciones o servicios de red, no es un estándar, sino un conjunto de principios tales como [18]:

- Sin Estado: No almacena ningún contexto o sesión del cliente en el servidor entre peticiones. La ausencia de sesiones de clientes elimina la necesidad de sincronizar datos de sesión con aplicaciones externas.
- Servicios Uniformes: Uso explícito de métodos HTTP por ejemplo: *GET*, *POST*, *PUT* y *DELETE*.
- Cliente-Servidor: Deben basarse en una arquitectura cliente-servidor, donde las responsabilidades se encuentren completamente separadas, el servidor contendrá los recursos y los estados de estos, mientras el cliente accederá a ellos.
- Caché: Deben tener la posibilidad de marcar las respuestas del servidor como cacheables, así al realizar la primera petición del servicio, las siguientes pudieran apoyarse en la caché si fuera necesario [18].

II.14 JWT (JSON Web Tokens)

Es el estándar usado para la autenticación sin estado con *Tokens*. Su funcionamiento consiste en generar un *Token* luego de que un usuario se autentica en la aplicación de esta forma cada petición HTTP que realice va acompañada del *Token* en la cabecera. Este *Token* es una firma cifrada que permite al APIRest identificar al usuario. El *Token* no se almacena en el servidor, sino del lado del cliente en el *localStorage* por ejemplo, desde el API se descifran los Token para redirigir el flujo de la aplicación en el sentido que lo requiera.

CAPÍTULO III – MARCO METODOLÓGICO

Es de vital importancia trabajar con una metodología que involucre al cliente, ya que para el sistema planteado en este trabajo instrumental de grado es necesaria su participación durante el desarrollo. Las metodologías ágiles presentan diversas ventajas haciéndolas una buena opción para el desarrollo indicado, entre ellas se puede destacar:

- Capacidad de respuesta a cambios de requisitos a lo largo del desarrollo.
- Entrega continua y en plazos breves de software funcional.
- Mejora continua de los procesos y el equipo de desarrollo.
- Trabajo conjunto entre el cliente y el equipo de desarrollo.

III.1. Crystal

Se trata de una familia de metodologías enfocadas en el desarrollo de proyectos, creadas por Alistair Cockburn, caracterizadas por estar centradas en las personas que componen el equipo y la reducción al máximo del número de artefactos producidos, haciendo énfasis en la comunicación y cooperación de la gente. Las metodologías *Crystal* permiten el uso de prácticas y herramientas de otras metodologías ágiles como XP o Scrum para la planificación e inicio del proyecto [3].

Esta familia de metodologías no especifica ningún ciclo de vida concreto, ni ningún modelo de procesos, en vez de eso lo que hace es determinar una guía de las políticas estándar, productos de trabajo,

herramientas, técnicas y roles. Las políticas que se implementen en la metodología dependerán del tamaño del equipo, proyectos más grandes suelen necesitar una mayor coordinación y metodologías más complejas que los pequeños. *Crystal* dispone un código de color para marcar la complejidad de una metodología, cuanto más oscuro es el color más pesado es el método, por ejemplo, *Crystal Clear* (hasta seis o posiblemente ocho miembros) o *Crystal Orange* (hasta cuarenta miembros). En la Figura 1 se muestra una evaluación de las pérdidas que puede ocasionar la falla de un sistema, se puede observar además los colores que representan cada metodología [21].

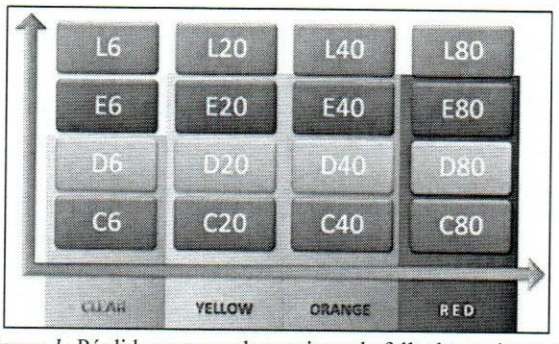


Figura 1: Pérdidas que puede ocasionar la falla de un sistema
Fuente [21]

En la figura 1 también se muestran dos flechas, la horizontal representa el tamaño del proyecto mientras que la vertical se refiere a la criticidad del sistema, los caracteres (C, D, E y L) representan los parámetros:

- C, indica pérdida de *confort* debido a un fallo en el sistema.
- D, indica pérdida de dinero discrecional, es decir del que se puede disponer.

- E, indica pérdida de dinero esencial, es decir dinero del que no se puede disponer libremente.
- L, de *life* en inglés, vida. Indica la pérdida de vidas por el fallo del sistema.

"Las dimensiones de criticidad y tamaño son representadas por un símbolo en la categoría. Por ejemplo D6 indica un proyecto con un máximo de seis personas, de máxima criticidad de dinero discrecional" [21]. La caída de un sistema que ocasiona incomodidades indica que su nivel de criticidad es C, mientras que si causa pérdidas de vida su nivel es L. Para el proyecto descrito en este trabajo instrumental de grado se determinó que su nivel de criticidad es C, es decir un fallo en el sistema ocasionaría pérdida de *confort*.

II.2. Crystal Clear

Es una metodología diseñada para proyectos no críticos, de poco tamaño y para una cantidad de máximo, 8 miembros del equipo.

III.2.1. Roles de Crystal Clear

- **Patrocinador:** Es quien tiene en mente la visión a largo plazo del proyecto y además es quien proporciona el dinero o busca los medios para conseguirlo.
- **Usuario Experto:** Es quien está familiarizado con los procedimientos operativos del sistema en uso, conociendo cuáles son los modos de operación más y menos frecuentemente utilizados. Se espera que conozca las reglas de negocio.

- **Diseñador principal:** Esta persona es el líder técnico, se supone que tiene experiencia en el desarrollo de software, es quien será capaz de realizar la mayor cantidad de diseño del proyecto. Se espera que desempeñe tareas tanto de diseño como de programación.
- **Diseñador-programador:** Produce junto con el diseñador principal, los borradores de pantallas, el modelo común de dominio, código fuente, pruebas y sistemas de empaquetado.
- **Experto de negocio:** Debe conocer las reglas y políticas de negocio.
- **Coordinador:** El coordinador puede ser un rol parcial que ocupe algún miembro del proyecto y es quien maneja y coordina el proyecto.
- **Verificador:** Es el encargado de realizar las pruebas [3].

II.2.2. Valores, propiedades y técnicas

Crystal contiene un conjunto de propiedades entre las cuales se pueden listar: entrega frecuente, mejora reflexiva, comunicación osmótica, seguridad personal, foco, fácil acceso a usuarios expertos y ambiente técnico con pruebas.

También dispone de las siguientes estrategias: exploración 360°, éxito temprano, esqueleto caminante, rearquitectura incremental y radiadores de información.

Las siguientes son técnicas que dispone *Crystal*: entrevistas de proyectos, talleres de reflexión, planeamiento Blitz, estimación Delphi,

encuentros diarios de pie, miniatura de procesos, gráficos de quemado, programación lado a lado.

A continuación, se listan las propiedades, estrategias y técnicas que serán aplicadas en este trabajo instrumental de grado:

Entrega frecuente: Consiste en entregar software a los clientes con frecuencia, no solamente en compilar código. La frecuencia dependerá del proyecto, pero puede ser diaria, semanal o mensual.

Mejora reflexiva: Tomarse un pequeño tiempo (unas pocas horas cada semana o una vez al mes) para pensar bien qué se está haciendo, cotejar notas, reflexionar, discutir.

Fácil acceso a usuarios: Una respuesta que no tarde tanto tiempo, es decir, desde el momento que surge una duda sobre el desarrollo de la funcionalidad se podrá contactar con el usuario experto para realizar las preguntas necesarias y continuar con el desarrollo. Esto permite que el usuario sea una parte más del equipo.

Exploración de 360º: Verificar o tomar una muestra del valor de negocios del proyecto, los requerimientos, el modelo de dominio, la tecnología, el plan del proyecto y el proceso.

Talleres de reflexión: El equipo toma alrededor de treinta minutos o una hora para discutir inconvenientes y mejoras así como planear el próximo incremento [3].

III.2.3. Justificación de selección

Crystal Clear es la metodología que se aplicó para el desarrollo del sistema presentado en este trabajo instrumental de grado. Se seleccionó por su enfoque ágil, y porque el proyecto es pequeño y de baja criticidad. A pesar de que *Crystal* está diseñada para grupos o equipos de desarrollo de uno a máximo ocho miembros, para el sistema de este proyecto sólo se contaba con un programador que es el autor de este trabajo, un tutor empresarial y algunos bionalístas e investigadores pertenecientes a la fundación, sin embargo al no ser un proyecto de gran envergadura como para justificar un equipo más grande y manejando los roles que dispone *Crystal Clear* se pudieron asignar los roles como se muestra en la tabla 1:

Tabla 1 *Roles de la metodología designados en el proyecto*

Nombre	Roles
Yaneliz (tutor empresarial)	Patrocinador
Bionalistas/Investigadores	Usuario experto, Experto de negocios
Mariangela	Diseñador Principal, Coordinador, Verificador, Diseñador – Programador.

Fuente: Elaboración propia

Otra de las razones que llevaron a su selección es su objetivo de enfocar al equipo en las características importantes y por ser una metodología fácil de adaptar, además, del gran énfasis que hace en la comunicación que es de vital importancia en este proyecto ya que se maneja data muy importante como el cálculo de volúmenes de los reactivos. Además, tiene un manejo liviano en relación a los entregables.

Crystal maneja iteraciones cortas con *feedback* frecuente por parte de los usuarios/clientes, minimizando de esta forma la necesidad de productos intermedios. Otro de los planteamientos es la necesidad de disponer de un usuario real, al menos de forma parcial, para realizar validaciones sobre las Interfaces del Usuario y además participe en la definición de los requerimientos funcionales del sistema, este factor fue de gran importancia para este proyecto ya que se involucró de forma importante a los bionalistas e investigadores que serán los usuarios finales del sistema.

III.2.4. Proceso

La familia de metodologías *Crystal* no especifica ningún ciclo de vida concreto, así como tampoco ningún modelo de procesos. Es por esto que tomando en consideración las prácticas más comunes de las metodologías *Crystal* como: planificación por etapas, revisiones y resúmenes, monitorización y paralelismo flujo, se planteó un proceso tomando como referencia el caso de estudio que Alistair Cockburn plantea en su libro [3].

En la Figura 2 se muestra el proceso implementado en este trabajo instrumental de grado.

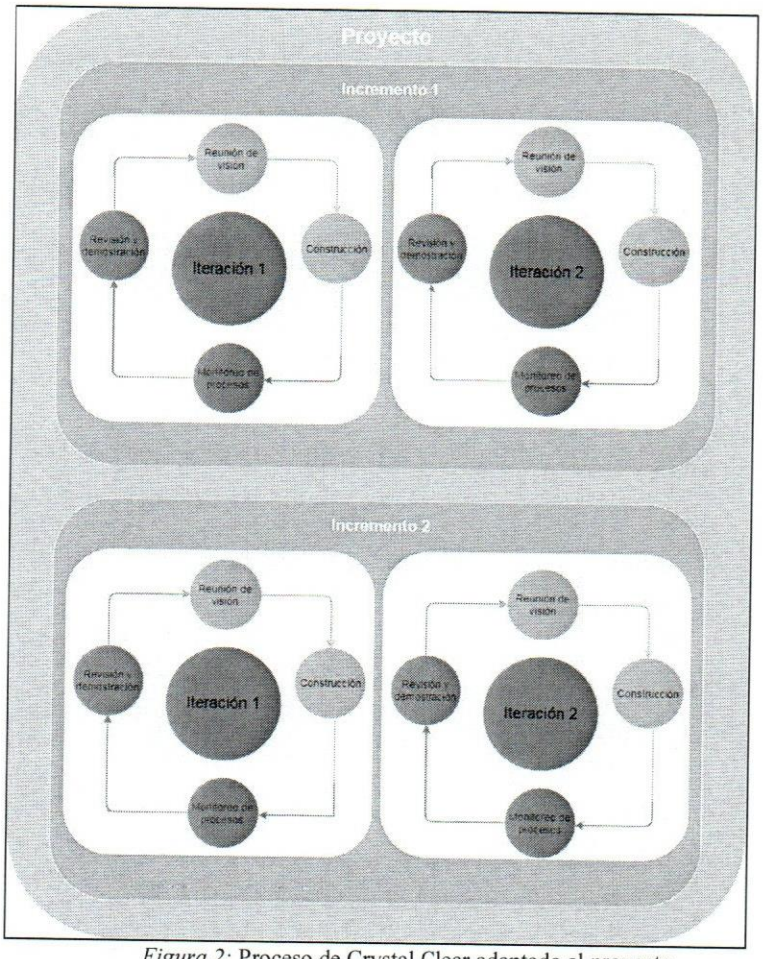


Figura 2: Proceso de Crystal Clear adaptado al proyecto
Fuente: Elaboración propia

En la implementación de esta metodología se tomó la idea de la "fase de exploración" de XP para convertirla en la primera fase del proyecto en la cual se tomarían decisiones como la selección de herramientas. En la tabla 2 se presenta cada una de los incrementos e iteraciones que conformaron el proyecto y el objetivo que se desarrolló en cada una.

Tabla 2 *Fases y actividades del proyecto*

Fases	Actividades
Fase de exploración	Levantamiento de información
	Elaboración de requerimientos funcionales
	Evaluación de las plataformas de desarrollo **(Aporte tecnológico)**
	Primera reunión de planificación
Incremento 1	- Iteración 1: Creación del modelo de base de datos que se adapte a la solución propuesta **(Objetivo 1)**.
	- Iteración 2: Creación de un módulo de roles, permisos y gestión de usuarios **(Objetivo 2)**.
Incremento 2	- Iteración 1: Creación de módulo inventario de insumos, reactivos y consumibles **(Objetivo 4)**.
	- Iteración 2: Creación de módulo de pruebas y análisis que aloje información sobre su implementación **(Objetivo 3)**.
Incremento 3	- Iteración 1: Creación de un módulo de cálculos y fórmulas matemáticas **(Objetivo 5)**.
	- Iteración 2: Creación de un módulo de registro de procesos de muestras **(Objetivo 6)**.
	- Iteración 3: Integración con el sistema de mantenimiento **(Aporte tecnológico)**.
Incremento 4	- Iteración 1: Creación de un módulo de alertas de fallos del almacenamiento **(Objetivo 7)**.
	- Iteración 2: Creación de un módulo de histórico de análisis **(Objetivo 8)**.
Incremento 5	- Iteración 1: Creación de un módulo de estadísticas **(Objetivo 9)**.

Fuente: Elaboración propia

CAPÍTULO IV – DESARROLLO

En este capítulo se describe detalladamente la implementación del sistema planteado como solución en este trabajo Instrumental de grado. Basado en el marco metodológico propuesto con *Crystal Clear*, se tomó en cuenta sus propiedades y el proceso planteado para el diseño y desarrollo del sistema, de acuerdo con eso se definieron cinco incrementos con duración de tres a cuatro semanas. A continuación se presentará este capítulo con todo el desarrollo del proyecto estructurado por incrementos e iteraciones.

IV.1. Fase de exploración

Aplicando el proceso planteado con *Crystal Clear*, se tomó la idea de la "fase de exploración" de la metodología XP para poder preparar los insumos necesarios para el inicio de los incrementos. Así pues, fue necesario realizar reuniones con el equipo de la Fundación Jacinto Convit para así poder definir una lista de requerimientos. En base a los requerimientos y su desglose en historias de usuario (Ver Apéndice C), se plantearon cinco incrementos con duraciones de tres a cuatro semanas, para cada uno hubo una cantidad de iteraciones que se fueron definiendo según la complejidad de los requerimientos y el tiempo estimado.

En esta fase de exploración se realizó el levantamiento de información sobre las actividades diarias que se realizaban en el laboratorio de la fundación para así poder tener más detalles sobre su funcionamiento. Se pudo identificar como la Fundación no contaba con una fuente centralizada de datos y además,

el manejo del cálculo de pruebas y manipulación del inventario durante estas eran procesos adicionales que requería una inversión de tiempo y en algunos casos interrumpía las actividades del personal del laboratorio.

Seguidamente se procedió a revisar la información recabada para abordar temas como la evaluación y selección de las tecnologías y herramientas que podrían usarse y así proceder a crear la arquitectura del sistema a desarrollar.

Se creó una visión general de cómo podría comportarse el sistema al rediseñar los procesos aplicados al análisis de muestras. En la Figura 3 puede observarse la interacción del sistema.

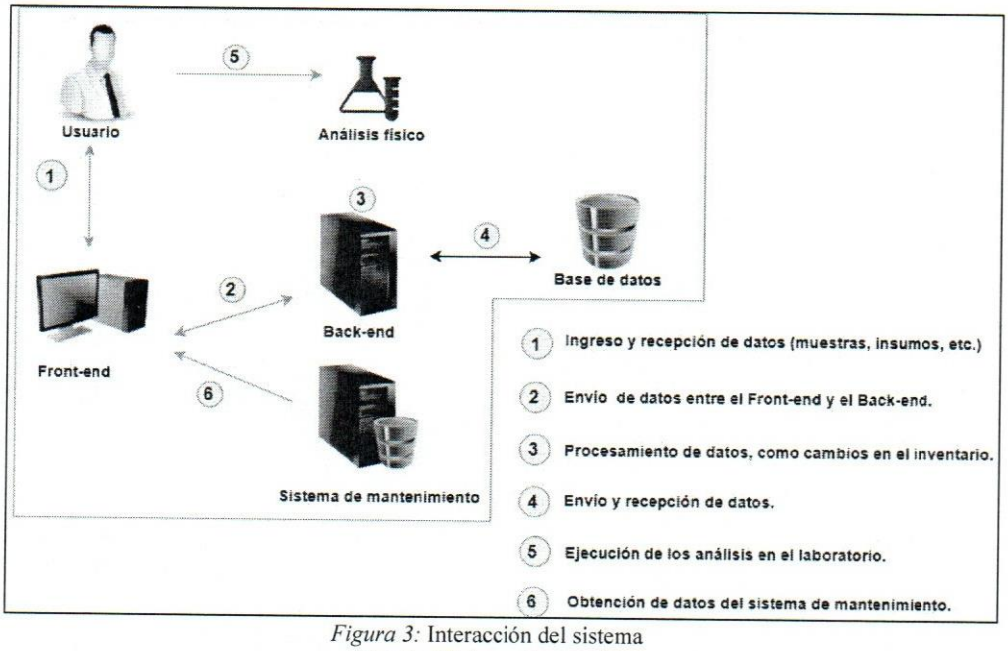


Figura 3: Interacción del sistema
Fuente: Elaboración propia

Al rediseñar el proceso en la aplicación de pruebas, los bionalistas e investigadores ingresarían en la sección de pre-análisis para agregar los datos de una nueva muestra y registrarla, dejando así la información centralizada y no distribuida en diferentes cuadernos o fichas. Posteriormente podrían pasar a la fase de análisis para aplicar las pruebas correspondientes. En el proceso original ellos debían buscar en diferentes fuentes cual era el procedimiento para aplicar la prueba y además realizar el cálculo manual para cada reactivo, según la cantidad de muestras a las cuales se les ejecutaría la prueba, ahora con el rediseño ya la información se encontraría centralizada y además los cálculos necesarios ya estarían siendo ejecutados por el sistema dentro de la fase de análisis, lo cual, hace más natural el proceso, enfocando el trabajo del personal del laboratorio, de igual forma se elimina el registro por parte del personal de la Fundación de las distintas cantidades de componentes e insumos utilizados en una prueba, el sistema haría ese trabajo sin que se necesite la intervención del usuario. Finalmente podrían realizar el post-análisis para registrar los resultados y/o diagnósticos encontrados.

Podría resumirse el proceso de la Figura 3 de la siguiente forma: una vez que el usuario ha ingresado en el sistema podrá registrar las muestras y sus órdenes médicas asociadas, ingresar en el inventario para agregar o quitar insumos, gestionar información de pruebas o ejecutar un proceso de análisis de muestras. La ejecución de cada prueba o actividad que se registra en el sistema, sería procesada en Back-End, finalmente el sistema almacenaría los cambios o nuevos registros en la base de datos.

IV.1.1 Evaluaciones

Al obtener los requerimientos funcionales fue necesario plantear qué herramientas serían óptimas para el desarrollo del sistema como parte del aporte tecnológico de este trabajo. En esta sección se da a conocer las evaluaciones que fueron realizadas en la selección de las herramientas.

Evaluación de la plataforma de desarrollo

Se tomaron en cuenta los lenguajes de programación más populares del momento según una encuesta realizada por *Stack Overflow* sobre los lenguajes más populares para el 2016 [19] así como también uno de los rankings más conocidos, el índice de comunidad de programación TIOBE [20] (El detalle de la encuesta se puede ver en el Apéndice D).

Al revisar las encuestas se puede observar que Java es uno los lenguajes más populares, esta fue una de la razones de su selección para la evaluación, además del conocimiento previo en el lenguaje que tenía el autor de este trabajo.

Para el momento en que se da inicio a este proyecto, se estaba popularizando el desarrollo en Java con el generador de código *JHipster*, así que se tomó la decisión de evaluar a través de la puesta en práctica si podría ser conveniente o no, usar un generador de código.

Un generador de código o generador de aplicaciones es una herramienta que tiene como finalidad agilizar el desarrollo a través de una descripción del modelo que se necesite crear (puede ser un modelado como UML) y de esa forma crear un porcentaje amplio de código.

JHipster: Es un generador de aplicaciones basado en *Yeoman*, que crea la base para continuar un proyecto basado en tecnologías como:

Del lado del cliente: *Responsive Web Design,* HTML, *Twitter Bootstrap, AngularJS.*

Del lado del servidor: Spring Boot, Spring Security, Spring MVC REST + Jackson, Soporte opcional WebSocket con Spring Websocket, Spring Data JPA + Bean Validation, Actualizaciones de base de datos con Liquibase, Soporte para Elasticsearch para búsqueda por encima de la base de datos, Soporte para MongoDB para base de datos NoSQL orientada a documentos. Soporte para Cassandra para base de datos NoSQL orientada a columnas. [8]

La versión que se usó para el momento de las pruebas fue *JHipster* 4.0.7 (al momento de escribir este trabajo instrumental de grado se encuentra en la versión 4.14.0). Para iniciar la prueba con el generador de código fue necesario tener un ambiente con *Java 8*, *Maven*, *Git* y *Node.js*. Además, era necesario instalar *Yeoman* quien sería el encargado de generar el *scaffolding,* también *Bower* quien facilita la instalación de librerías, *Gulp* que automatiza tareas comunes de desarrollo y finalmente *JHipster*.

Al usar el comando `yo jhipster` desde la consola, se muestran una serie de preguntas para configurar la aplicación como por ejemplo el tipo de aplicación que se desea crear Monolítica u orientada a microservicios, el nombre de la aplicación, el tipo de autenticación que se desea usar como *HTTP Session* o *JWT authentication*, el tipo de base de datos a usar, entre otras. La Figura 4 muestra parte de esta información.

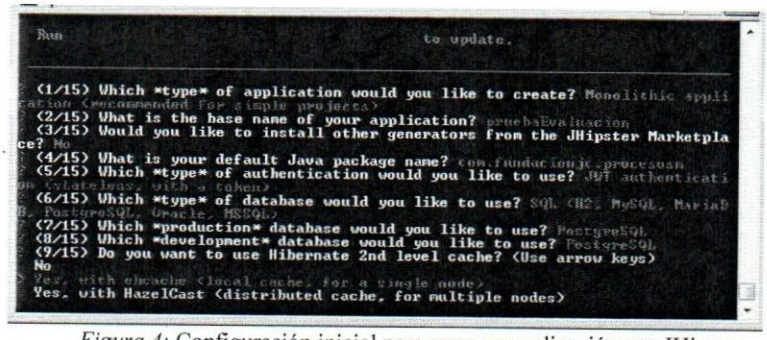


Figura 4: Configuración inicial para crear una aplicación con JHipster
Fuente: Captura realizada por el autor de este trabajo durante las pruebas de desarrollo.

Al finalizar se generó toda la estructura del proyecto, con algunas entidades como Usuario, vistas ya implementadas como el registro de usuarios y el inicio de sesión.

Jhipster también pone a disposición una herramienta *web* llamada *JDL Studio* que permite crear un esquema de entidades y que al finalizarlo puede ser exportado en un archivo que es leído desde la consola y *JHipster* genera las vistas y métodos básicos. En la Figura 5 se muestra un diagrama de prueba creado desde la herramienta.

Al realizar la importación del archivo, JHipster creó las vistas correspondientes a cada entidad y sus funcionalidades básicas (crear, consultar, actualizar y eliminar).

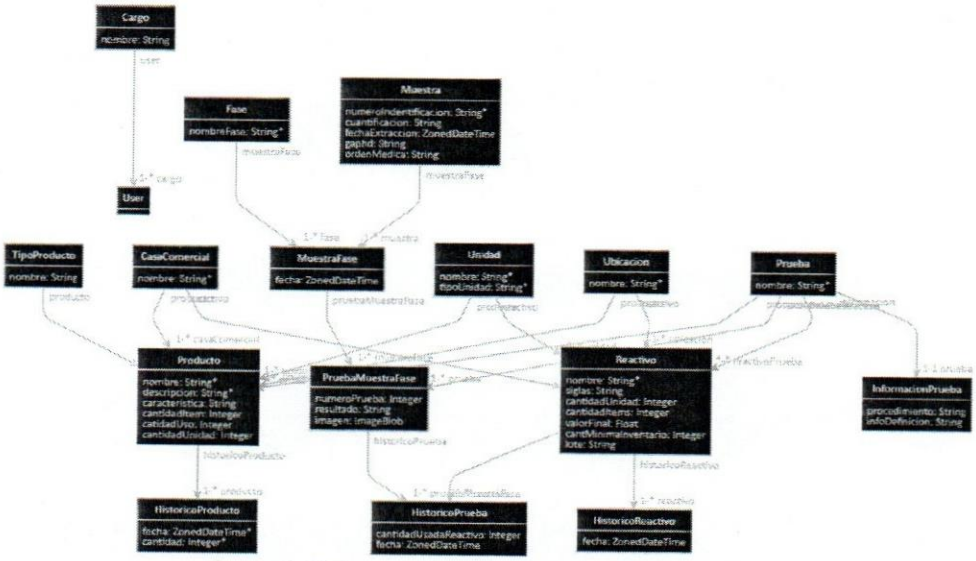


Figura 5: Diagrama del proyecto creado con JDL Studio
Fuente: Elaboración propia

Ventajas de usar un generador de código:

- ✓ Ahorro del tiempo en la configuración inicial del proyecto.
- ✓ Agiliza el desarrollo en un principio al crear un proyecto funcional que puede ser mostrado al usuario, puede resultar muy conveniente para crear prototipos funcionales.
- ✓ La tecnología que usa *JHipster* se encuentran entre las más actuales del momento.

Desventajas al usar un generador de código:

- De la gran cantidad de código que se genera, hay secciones que no suelen ser útiles.
- La inclusión de nuevas funcionalidades en cierto punto del desarrollo puede llegar a ser engorroso.

- Los cambios sencillos pueden llegar a convertirse en tediosos o complicados.
- La mantenibilidad del código puede resultar compleja.

Conclusión:

En el uso de *JHipster* se evidenció que a través de un conjunto de comandos se generaba una aplicación básica que ya tenía implementados el inicio de sesión y la generación de próximas vistas podía realizarse al importar un diagrama de clases creado con la herramienta *JDL Studio*.

A pesar de que los generadores de código pueden ahorran tiempo en el inicio del desarrollo de un proyecto, no siempre suelen ser favorables para la organización por la cantidad de código que se genera. Tomando en cuenta criterios como la mantenibilidad, que al generarse gran cantidad de código puede no ser tan mantenible en el tiempo y la escalabilidad, si la Fundación tomara la decisión de seguir ampliando el proyecto, se decidió no usar un generador de código, sin embargo, se tomaron algunas de las tecnologías usadas del lado del servidor para el desarrollo del proyecto.

A continuación se listan las herramientas seleccionadas:

Back-End: Entre las herramientas seleccionadas para el *back-end* está el *framework* Spring Boot, además para el manejo de datos y relaciones se usó Spring Data JPA.

Front-End: La encuesta de *Stack Overflow* situaba a *JavaScript* como uno de los lenguajes más populares, así como a *TypeScript* dentro de los más

deseados (Ver Apéndice D). Es por esto que dado el surgimiento de diferentes *frameworks* para *JavaScript* se selecciona dentro de los más populares y en avance a Angular.

Evaluación para la selección del lenguaje orientado a los cálculos matemáticos

Antes de realizar la evaluación a estos lenguajes, se hizo una revisión en detalle de los cálculos y fórmulas que se requerían en la aplicación de las pruebas que se realizan en los laboratorios de la Fundación. De esta revisión, se obtuvo que una de la operaciones más necesarias era el cálculo del volumen inicial de un reactivo, siendo este un problema de proporcionalidad, se requería aplicar una regla matemática, llamada regla de tres, además de algunas operaciones matemáticas básicas no se encontraba la necesidad de agregar un lenguaje extra a la solución que además tuviese que ser integrado con la plataforma de desarrollo seleccionada en la evaluación anterior (Ver apéndice F para detalles de los cálculos), sin embargo se quiere dejar reflejado en esta sección un par de lenguajes que fueron pensados en un inicio para la resolución de cálculos matemáticos, ellos son Octave y Julia:

- Octave: Es un lenguaje interpretado orientado a la matemática, además es software libre. Es considerado un lenguaje equivalente a *MatLab* pero en una versión libre [12].
- Julia: Es un lenguaje de programación creado por Stefan Karpinski. El lenguaje es conocido en el ámbito científico por su variedad de

funciones matemáticas, a la vez es un lenguaje de alto nivel y de alto *perfomance*. La librería de funciones matemáticas, en gran parte está escrita en Julia, pero también se integra con librerías desarrolladas en lenguaje C [7].

IV.1.2 Arquitectura

El propósito de plantear una arquitectura es definir la estructura en la que se podrá desarrollar el sistema web con cada una de sus funcionalidades que estará compuesto de elementos con propiedades y relaciones entre ellos.

La arquitectura que fue creada para el sistema web presentado en este TIG se dividió en dos grandes capas: *Front-End* y *Back-end*, con el objetivo de independizar la lógica de la vista. A continuación se detalla cada una de las capas y sub-capas:

- ***Back-End***: Esta capa contiene toda la lógica del negocio. Aplicando la estructura que provee el *framework* Spring Boot, el cual se basa en el estilo de arquitectura *REST (Representational State Transfer)*. Las siguientes son las capas pertenecientes al back-end:
    - Dominio: Contiene los objetos pertenecientes al dominio del negocio, como por ejemplo las entidades.
    - Repositorio (*Repository*): Es la capa donde se almacenan las clases que sirven de intermediarias entre el acceso a datos y la capa de lógica, es decir en esta se encuentran las interfaces pertenecientes al *CRUD (Create, Read, Update and Delete)*. Para esta capa se

trabajó con las herramientas de repositorio que provee el *framework* de Spring Data, al cual se puede acceder desde una invocación de los paquetes iniciales de Spring Boot.

- Lógica de Negocio: Es la capa donde se encuentra toda la implementación de las reglas de negocio y los servicios del sistema. Se comunica con la capa de controladores Web Rest.
- Web Rest: Es la capa en la que se encuentran las clases controladoras pertenecientes a los servicios REST. Esta capa actúa como intermediario entre el *Front-End* y la sub-capa de la lógica de negocio, ya que a través de servicios web se comunica con *el Front-End* para proveerle los datos que sean necesarios y contiene todas las invocaciones a métodos que están desarrollados en la sub-capa lógica de negocio.

- ***Front-End***: Esta capa engloba todo el manejo de interfaces y componentes con los que interactuaría el cliente, todo lo que visualice el usuario del sistema. Se comunica con los servicios web de la capa *Back-End*, quienes le proveen la información que necesita ser mostrada al usuario.

Se utilizó la arquitectura que brinda el *framework* Angular. La Figura 6 muestra la arquitectura de Angular.

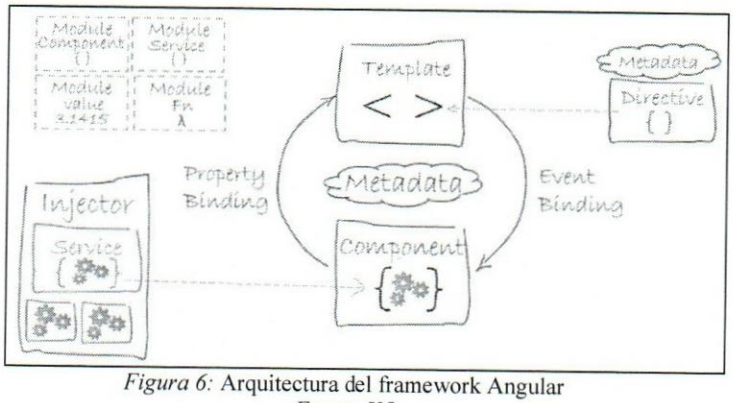


Figura 6: Arquitectura del framework Angular
Fuente [2]

Las aplicaciones creadas en Angular son modulares. Cada pantalla en angular deberá tener los siguientes componentes generales:

- Pantalla/*Template*: Representa la interfaz visual de interacción con el usuario, apoyado en el *framework* Boostrap, simplificando la creación y manipulación de los estilos y visualización de contenido.
- Componente: Contiene la lógica de las validaciones que se realizaron en los campos o componentes del *template* al que hace referencia así como los diferentes servicios que necesita.
- Servicios: Es la capa encargada de la comunicación con el *Back-end*, por lo tanto contiene todas las llamadas a los servicios *web*.

La Figura 7 muestra la arquitectura general del sistema.

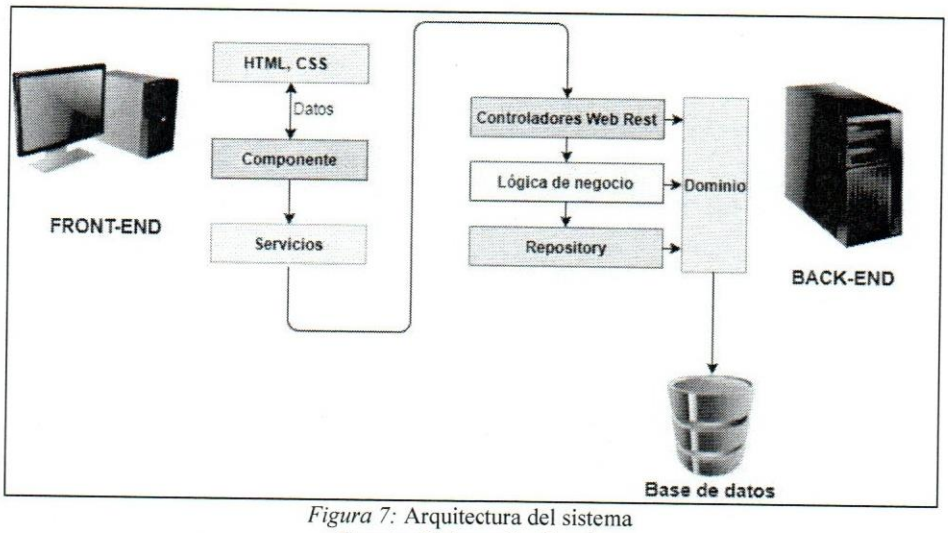


Figura 7: Arquitectura del sistema
Fuente: Elaboración Propia

IV.1.3 Usuarios

El sistema *web* fue desarrollado para su uso en los laboratorios de la Fundación Jacinto Convit, con el objetivo de mejorar sus procesos medulares y servir de apoyo durante la realización de las pruebas que se efectúan a una muestra. El grupo de usuarios que ingresan al sistema se divide principalmente en dos grupos, Administradores e Investigadores, además de estos dos roles también existe un consultor quien tendrá acceso limitado.

- Investigador: Pueden acceder a casi todas la opciones disponibles en el sistema web, estos usuarios pueden consultar el módulo de inventario y el histórico de análisis. En el módulo de información de pruebas, los investigadores podrán gestionar los datos de esta sección, ya sea editando o agregando nuevos aportes; Una de las principales actividades a las que podrá acceder este usuario, es al módulo de

procesamiento de muestras, así como también al módulo de cálculos matemáticos.

- Administrador: Los usuarios administradores son los únicos que pueden crear usuarios nuevos y asignarles la permisología correspondiente. Los administradores también podrán realizar todas las actividades a las que tiene acceso un usuario Investigador.
- Consultor: Estos usuarios son los más limitados del sistema, el grupo de actividades que pueden realizar solo están dirigidas al módulo de información de pruebas, por lo tanto podrán agregar, editar o eliminar datos de la *wiki* y al módulo del histórico de análisis, el cual es solo de consulta.

IV.2. Incremento 1

Para el primer incremento fue necesario realizar el diseño de un modelo de datos que contemplara todos los requerimientos funcionales basados en el levantamiento de información. La implementación de la base de datos dio inicio al desarrollo del sistema y se requirió una sección en la que se gestione las cuentas de usuarios y sus permisos dentro del sistema.

IV.2.1. Iteraciones

- Creación del modelo de base de datos que se adapte a la solución propuesta.
- Creación de un módulo de roles, permisos y gestión de usuarios.

IV.2.1.1 Modelo de base de datos

Para dar inicio al desarrollo del sistema fue necesario realizar el diseño de un modelo entidad-relación (E-R) basado en los requerimientos obtenidos en las reuniones con el equipo de la Fundación Jacinto Convit (listado de requerimientos en Apéndice C). Esta lista de requerimientos funcionales permitió la elaboración de un modelo entidad relación que puede observarse en el Apéndice E. La creación del modelo fue realizada de tal forma que permitiera ajustes en los incrementos posteriores a medida que se fueran detallando los requerimientos de cada iteración.

Por petición de la fundación el manejador de base de datos que se usaría es *PostgreSql*. La creación de estas estructuras, permitiría el almacenamiento de muestras, información detallada de los análisis, llevar el almacenamiento de los insumos y productos, así como también datos de los usuarios registrados en el sistema; Al finalizar la creación del modelo E-R, se procedió a la creación de los objetos de negocio, los objetos de negocio fueron creados en el *Back-End*, específicamente en la capa de Dominio usando *Spring Data JPA* el cual facilita la manipulación de la base de datos a través de objetos.

IV.2.1.2 Modelo de roles, permisos y gestión de usuarios

Para manejar la permisología en el sistema fueron destinadas las tablas de usuarios y roles en la base de base de datos.

La gestión de usuarios dentro del sistema solo puede ser manipulada por Administradores, por lo tanto se procedió a la creación de una vista de registro de usuarios, creando un formulario con los campos de texto, como nombre, cédula y demás campos necesarios para el registro, además es desde esta ventana donde el administrador asignará el rol que tendrá el usuario dentro del sistema, los cuales pueden ser: "Administrador", "Investigador" y "Consultor".

Un administrador deberá ingresar en el menú de "Administración" (la Figura 8 muestra el menú de opciones del sistema) en la opción "Lista de usuarios", desde allí se despliega la vista que muestra los usuarios registrados del sistema, para agregar uno nuevo se deberá acceder en la opción de Registro e ingresar todos los datos requeridos (ver Apéndice F.1).

Figura 8: Menú de opciones del sistema
Fuente: Elaboración propia

La interfaz que lista los usuarios ofrece servicios de consulta de los mismos por un administrador donde puede desactivar la cuenta de algún usuario sea requerido (ver Apéndice F.2).

- **Inicio de sesión:** Para el manejo de acceso al sistema se decidió usar la autenticación basada en *tokens* bajo el estándar JWT (*JSON Web Token*).

  Desde el *Fron-End* se creó un módulo con Angular para el ingreso al sistema, la vista de "Inicio de sesión" contendría los campos de

"usuario" y "contraseña" que deberán ser llenados por el usuario que requiera entrar en el sistema, (Ver Apéndice F.3). Si un usuario ha olvidado la contraseña podrá ingresar en la opción "¿Olvidaste tu contraseña?" para recuperarla, deberá ingresar su dirección de correo electrónico en el campo de texto que se muestra, el sistema enviará un correo y el usuario deberá acceder al enlace adjunto en el texto del *mail*, de esta forma entra en la pantalla de restablecer contraseña (Ver Apéndice F.4), en la cual solo deberá ingresar una nueva contraseña.

IV.3. Incremento 2

Para el segundo incremento se trabajó en la creación del módulo de inventario de reactivos y productos que son usados a diario en las instalaciones de la Fundación, para el área de Investigación y Diagnóstico Molecular. También se creó el módulo que provee información de las pruebas que se realizan en la Fundación, es allí donde se almacenan los protocolos e insumos necesarios en la ejecución del análisis.

IV.3.1 Iteraciones

- Creación de módulo inventario de insumos, reactivos y consumibles.
- Creación de módulo de pruebas y análisis que aloje información sobre su implementación.

IV.3.1.1 Módulo de inventario de insumos, reactivos y consumibles

Para el desarrollo de este módulo se creó una nueva categoría en el menú principal de la aplicación llamado "Inventario" el cual contiene un sub-menú con las opciones "Producto" y "Reactivo". La decisión de tener estas dos opciones provino del equipo de la Fundación, quienes requieren tener dividido el inventario en dos secciones. Antes de iniciar el desarrollo de producto y reactivo fue necesario realizar la implementación de funcionalidades para "Casas comerciales", "Tipos de productos", "Unidades" y "Ubicaciones", las cueles forman parte de la información que necesitará un reactivo o producto para ser registrado en el sistema.

1. **Casas comerciales**: Incluye las operaciones básicas (crear, modificar y activar/desactivar) esto para gestionar las marcas o fabricantes.
2. **Tipos de producto**: Incluye las operaciones de crear y modificar tipos de productos. Permitirá asociar los productos que se manejen en la aplicación.
3. **Unidades**: Incluye las operaciones de crear y modificar para gestionar las diferentes unidades que se manejan tanto para los reactivos como las unidades de contenidos de productos.
4. **Ubicaciones**: Incluye las operaciones de crear y modificar para gestionar ubicaciones como los laboratorios, estantes o neveras.

Algunas de las vistas se encuentran en el Apéndice F.5.

Un usuario con permisos de gestión de inventario, podrá consultar los productos o reactivos que se encuentren en existencia (Ver Apéndice F.6).

- Creación de un producto/reactivo: Para ingresar a las vistas que permiten la creación de productos o reactivos, se añadió un botón de agregar producto en la ventana de consulta del inventario para productos. En el caso de los reactivos también se añadió su respectivo botón para la creación. En ambos caso se diseñó e implementó una vista que contiene un formulario con los datos que deben ingresarse para realizar el registro de un producto o de un reactivo, campos tales como nombre, característica, presentación entre otros, forman parte de un producto. En el caso de un reactivo se requerirán datos como nombre, siglas, cantidad por unidad entre otros. (Pantalla de registro de productos ver Apéndice F.7, pantalla de registro de reactivos ver Apéndice F.8).

IV.3.1.2 Módulo de pruebas y análisis que aloje información sobre su implementación

Para el desarrollo de este módulo se creó un nuevo *item* en el menú principal de la aplicación llamado "Wiki". Al ingresar al sub-menú "Pruebas" se despliega una ventana que muestra el listado de todas las pruebas que han sido registradas. El registro de información para una nueva prueba se realiza al seleccionar la opción de agregar, presentando un formulario de datos de ingreso: nombre de la prueba, los reactivos y productos que son necesarios para la aplicación del análisis; de esta forma se crean los vínculos entre una prueba y producto, así como entre prueba y reactivo (Ver Apéndice F.9), estos

insumos se agregan desde un componente de selección múltiple, las cantidades requeridas para cada prueba son ingresadas desde una segunda vista donde se mostrarán los productos y reactivos seleccionados del componente de múltiple selección

IV.4. Incremento 3

El desarrollo de este incremento se dedicó al diseño e implementación de un módulo que permite automatizar los cálculos necesarios para ejecutar una prueba, además un módulo que contempla las tres fases pertenecientes al proceso de análisis de muestras, agregando en la segunda fase la integración con el sistema de mantenimiento de la Fundación.

IV.4.1. Iteraciones

- Creación de un módulo de cálculos y fórmulas matemáticas.
- Integración con el sistema de mantenimiento.
- Creación de un módulo de registro de procesos de muestras.

IV.4.1.1. Módulo de cálculos

Para el desarrollo de este módulo se definieron los tipos de cálculos que se manejan en el laboratorio. Para el equipo de la Fundación es necesario que antes de ejecutar una prueba se obtenga por ejemplo, el volumen específico de un reactivo, siendo este un problema de proporcionalidad, se requiere aplicar una regla matemática, llamada regla de tres.

A partir de lo indicado anteriormente, se creó una sección donde los usuarios pueden a modo de práctica, generar los volúmenes necesarios de un grupo de reactivos pertenecientes a una prueba, y que físicamente después deberán mezclarse para ejecutar el análisis. Por ejemplo, la PCR es una de las pruebas que más se aplican dentro de los laboratorios de la Fundación. Para realizar este análisis se necesita de reactivos como cloruro de magnesio (MgCl2), el cual de fábrica puede poseer una concentración (las concentraciones puede variar por fabricante) de treinta milimolar (30 mM) y del que se requiere una concentración final de veinticinco milimolar (25 mM) con un volumen final de reacción de veinte micro litros (20 uL), entonces será necesario calcular el volumen inicial del reactivo en solución. (Esquema de condiciones para la PCR, ver Apéndice G).

Para la creación de esta vista se incluyeron campos de texto que representan el volumen final de la prueba y las concentraciones inicial y final de los reactivos, estos campos son editables. Al ingresar en la ventana, el usuario deberá seleccionar un *item* de una lista desplegable que contiene las pruebas que se aplican en la Fundación y que han sido cargadas anteriormente desde el módulo de información de pruebas. Una vez seleccionada, se presentará la lista de los reactivos y las concentraciones que previamente pudieron haber sido almacenadas durante el registro del reactivo, las cuales son necesarias para poder aplicar el cálculo, y los usuarios podrán ingresar un número de muestras para que posteriormente el sistema pueda

emitir el cálculo de la *premix* o mezcla de reacción para el número deseado de análisis en pantalla. (Ver Apéndice F.10)

El módulo creado en este menú, tiene la finalidad de servir como práctica para la ejecución de cálculos, ya que es durante el proceso de análisis de muestras donde se asociarán los análisis a las muestras y se realizará la disminución en el inventario de los insumos usados.

Una parte de la implementación de este módulo se agregó también al registro de prueba del módulo de *wiki* de esa forma pueden registrarse los cálculos de los volúmenes de los reactivos asociados a una prueba y que puedan ser cargado para la ejecución de los análisis.

IV.4.1.2. Integración con el sistema de mantenimiento

El sistema de gestión de mantenimiento que posee la Fundación, almacena toda la información de uso de los equipos científicos, el principal objetivo de realizar una integración fue evitar la duplicidad en la data que se está gestionando. Para esta integración se agregó un nuevo servicio en Angular que realizara la invocación al API REST perteneciente al servicio de consulta de equipos del sistema de mantenimiento, el listado de datos obtenidos se incluirían en el módulo de pruebas en la fase de post-análisis, una sección para listar datos de los equipos pertenecientes a la Fundación.

IV.4.1.3. Módulo de registro de procesos de muestra

Para la Fundación es de vital importancia reflejar el proceso de estudio de muestras en tres etapas, por lo tanto este módulo se dividió en tres fases: pre-análisis, análisis y post-análisis, y se creó un menú llamado "Muestras" con un sub-menú que contiene cada etapa:

- **Pre-análisis:** Al ingresar en esta opción el usuario deberá llenar un formulario con campos como fecha de extracción, cuantificación, entre otros, y así registrar las muestras que ingresan en el laboratorio de la Fundación (Ver Apéndice F.11).
- **Análisis:** Fase principal dentro del proceso, donde se realizan los cálculos de volúmenes de cada uno de los reactivos que componen la prueba que se necesite ejecutar a una cantidad de muestras. Al ingresar en el sub-menú "Análisis" se muestra una vista que contiene dos secciones, en la primera sección se deberán agregar las muestras (que han sido registradas en la fase de pre-análisis) y a las que se les realizará el estudio.

  En la segunda sección se deben realizar los cálculos, este segmento es muy parecido al módulo de cálculos explicado anteriormente, el funcionamiento es el mismo, con la diferencia de que al realizar el cálculo el sistema se conecta al módulo de inventario y disminuye las cantidades de reactivo, y productos usados. (Ver Apéndice F.12).

- **Post-análisis:** Registrar el final del proceso de pruebas a una muestra, realizando el registro del diagnóstico o resultado, además, de una imagen que el personal de la Fundación deberá cargar desde el sistema. Dicha imagen muestra el resultado de la prueba. También se incluyó una lista desplegable donde se podrá seleccionar el equipo perteneciente a la Fundación donde se ejecutó el estudio, esta información es la obtenida de la integración con el sistema de mantenimiento. De ser necesario en esta fase se puede desactivar la muestra para que no se liste nuevamente en la fase de análisis. (Ver Apéndice F.13).

El proceso de pruebas hacia una muestra podrá consultarse desde el módulo histórico de análisis.

IV.5. Incremento 4

En el desarrollo de este cuarto incremento se definieron las notificaciones que estaría recibiendo el equipo de biólogos de la Fundación sobre la disminución de productos o reactivos dentro del inventario, así mismo se planteó cuáles debieron ser los parámetros de entrada y los datos de salida para listar el histórico de análisis.

IV.5.1 Iteraciones

- Creación de un módulo de alertas de fallos del almacenamiento.
- Creación de un módulo de histórico de análisis.

IV.5.1.1. Módulo de aletas de fallos del almacenamiento

Para este módulo se sostuvo una reunión con el equipo de la Fundación para establecer el tipo de notificaciones que se enviarían a través de correo electrónico. Se requería que cada vez que un producto o reactivo disminuyera en el inventario, el sistema pudiese notificar sobre aquellos producto que comienzan a faltar.

Para poder realizar una notificación de falla, fue necesario agregar un campo en las tablas de base de datos que almacene la cantidad mínima que debe tener un producto o reactivo para ser considerado como faltante, en el sistema, este dato es registrado desde el formulario de creación de un producto o de un reactivo, estableciendo el indicador de cuándo debe considerarse un insumo como faltante. Se desarrolló una vista a la cual se accede en el menú de Administración, sub-menú "Notificaciones", visualizando una lista con los correos de los usuarios pertenecientes al sistema y la notificación asignada que pueden recibir, un usuario administrador podrá agregar o eliminar correos a la lista de envío. (Ver Apéndice F.14).

IV.5.1.2. Módulo de histórico de análisis

Se diseñó y desarrolló una vista con un histórico de pruebas o análisis que se realizan a las muestras que ingresan en los laboratorios de la Fundación, a través, de la disposición horizontal de elementos, se agregaron dos campos de búsqueda: código de muestra y un rango de fecha (inicio y fin). Esta vista genera una sección en la cual se visualiza las muestras y las fases

en las que se encuentren (Ver Apéndice F.15). Se muestra información en detalle para cada muestra al generar un evento selección en el campo de código de muestra, el cual abrirá un *modal* con la información relevante.

Es imperativo haber iniciado la sesión del usuario dentro del sistema para poder ingresar al histórico de análisis dividido en las fases de pre-análisis, análisis y post-análisis; y en cada una de ellas, las muestras. También se muestran los campos de código de la muestra y rango de fecha (inicio y fin), el usuario podrá ingresar alguno de los datos y presionar el botón de búsqueda, a partir de esa acción se refresca la vista con los resultados. Los datos de la muestras pueden percibirse al ser seleccionado sobre el identificador de la mismas apareciendo una ventana emergente con datos relevantes de información como la fecha de ingreso de la muestra, la orden médica, experimentos aplicados, resultados entre otros. (Ver Apéndice F.16)

IV.6. Incremento 5

En este quinto incremento se analizaron las estadísticas que el equipo de la Fundación solicitaba, con el fin de diseñar las vistas necesarias para el sistema web.

IV.6.1. Iteraciones

- Creación de un módulo de estadísticas

IV.6.1.1 Módulo de estadísticas

Para el desarrollo de este módulo se creó un nuevo ítem en el menú dentro de "Wiki", llamado "Estadísticas". Se desarrolló una vista en la cual los usuarios pueden seleccionar la información que desean consultar:

- Cantidad de muestras y pruebas que se realizan con el contenido de un reactivo.
- Productos más usados por periodo de tiempo.
- Reactivos más usados por periodos de tiempo.
- Total de muestras y pruebas por periodo de tiempo.

Se trabajó con la librería *ng2-charts* para poder mostrar distintos tipos de gráficos correspondientes a la consulta realizada. Para el cálculo de cada consulta se realizaron funciones de base de datos bajo el lenguaje que *PostgreSql* tiene disponible el cual es *plpgsql*. Para la vista, se procedió a crear dos secciones dentro de la ventana, en la primera sección se selecciona el tipo de información a consultar y se agrega los parámetros necesarios como por ejemplo un rango de fecha. En la segunda sección mostrará el gráfico con la información solicitada por el usuario (Apéndice F.17).

CAPÍTULO V - RESULTADOS

En este capítulo se especifica los resultados obtenidos para cada uno de los objetivos propuestos en este Trabajo Instrumental de Grado en el cual se desarrolló un Sistema de gestión para procesos medulares en el estudio de muestras para la unidad diagnóstico molecular de la Fundación Jacinto Convit. A continuación se exponen los resultados obtenidos por cada uno de los objetivos específicos planteados:

- **Diseñar e implementar un modelo de base de datos que se adapte a la solución propuesta.**

Se obtuvo un modelo Entidad-Relación a partir de los requerimientos levantados a través de continuas reuniones que se manejaron con el equipo de la Fundación. En este modelo de datos quedaron representada las entidades y relaciones del sistema. Con la implementación se obtuvo un almacenamiento de muestras y las pruebas que se les realiza, información sobre los protocolos de las pruebas, almacenamiento de un inventario de productos y reactivos, así como también los usuarios que maneja el sistema. (Ver apéndice E).

- **Diseñar e implementar un módulo de roles, permisos y gestión de usuarios.**

Fue diseñado y desarrollado un módulo en el primer incremento, para la gestión de permisología y el registro de usuario, de esta forma los administradores del sistema pueden registrar usuarios y asignarles un rol

conveniente para las actividades que deba realizar dentro del sistema. También durante este incremento se obtuvo el inicio de sesión de usuarios y recuperación de su contraseña. (Ver apéndices F.1 – F.4).

- **Diseñar e implementar un módulo inventario de insumos, reactivos y consumibles.**

Se desarrolló un módulo de inventario de insumos durante el segundo incremento, con el fin de tener un control de los productos y reactivos que maneja actualmente la Fundación Jacinto Convit. Durante su desarrollo se implementó las funcionalidades necesarias y sus respectivas interfaces gráficas. Este inventario puede ser alterado por dos vías, la primera ingresando directamente al módulo y modificando las cantidades de los insumos, la segunda es a través del módulo de pruebas al realizar la fase de análisis. (Ver apéndices F.5 – F.8).

- **Diseñar e implementar un módulo de pruebas y análisis que aloje información sobre su implementación.**

Se diseñó e implementó un módulo que pudiese almacenar y mostrar información conceptual sobre las pruebas que la Fundación Jacinto Convit aplica actualmente. En este módulo se registran los protocolos de los análisis y los diferentes productos y reactivos con sus concentraciones, necesarios para la aplicación de cada prueba. (Ver apéndices F.9).

- **Diseñar e implementar un módulo de cálculos y fórmulas matemáticas.**

Se desarrolló un módulo para realizar los cálculos necesarios antes de la ejecución de una prueba. Este módulo es usado por el personal del laboratorio como parte de sus prácticas antes de la ejecución de una prueba, puede ser de gran utilidad para el personal nuevo que aún no se encuentren familiarizados con el sistema. (Ver apéndice F.10).

- **Diseñar e implementar un módulo de registro de procesos de muestras.**

Se desarrolló un módulo que cubriera el proceso de ejecuciones de pruebas a las muestras que ingresan en los laboratorios de la Fundación. El módulo está dividido en tres vistas que representan cada una de las fases del proceso, las cuales corresponden a:

Pre-análisis: En esta vista se registran las muestran que ingresan a la Fundación. (Ver apéndice F.11).

Análisis: En esta vista se realiza la ejecución de la prueba dentro del sistema. El usuario selecciona las muestras a la que se les debe aplicar el análisis, posteriormente selecciona una prueba que despliega los reactivos necesarios en esta y el usuario podrá generar los cálculos necesarios para la ejecución física de las pruebas. (Ver apéndice F.12).

Post-análisis: En esta vista el usuario realizará el registro del resultado y/o diagnóstico y guardará una imagen que corresponde a la prueba. También

se registra el equipo de la Fundación donde se realiza el análisis en físico. (Ver apéndice F.13)

- **Diseñar e implementar un módulo de alertas de fallos en el almacenamiento.**

Se diseñó e implementó un módulo para realizar notificaciones a través de correo electrónico cuando un insumo llega a la cantidad mínima establecida para ser tomado en cuenta para reposición. En el módulo se puede configurar los usuarios que pueden recibir estas notificaciones. (Ver apéndice F.14).

- **Diseñar e implementar un módulo histórico de análisis.**

Se diseñó e implementó un histórico de análisis que sirvió como página de inicio, de esta forma los usuarios que ingresen en el sistema podrán visualizar las pruebas que se están realizando, así como también pueden cambiar los rangos de fechas (inicio y fin) y realizar una nueva consulta. La información de las muestras también se visualiza desde este módulo. (Ver apéndice F.15 – F.16).

- **Diseñar e implementar un Módulo de estadísticas.**

Se diseñó e implementó un módulo de estadísticas, que permite visualizar información como los productos o reactivos usados en un periodo de tiempo seleccionado por el usuario, o la cantidad de muestras y pruebas que pueden aplicarse con el contenido de un reactivo. Los datos son mostrados en pantalla a través de diferentes gráficos. (Ver apéndice F.17).

- **Integrar el sistema web con el sistema de gestión de mantenimiento.**

Se integró el sistema de mantenimiento con el sistema de gestión para los procesos medulares de la Fundación. En el módulo de procesamiento de pruebas, la integración permite obtener los datos de los equipos de la Fundación para así mostrarlos en la fase de post-análisis. (Ver pantalla de post-análisis en el apéndice F.13).

- **Rediseñar los procesos medulares actuales aplicados al análisis de muestras.**

Se obtuvo un rediseño del proceso que se realizaba en la fundación para la aplicación de pruebas. La Fundación cuenta ahora con un proceso estructurado y centralizado para la gestión de información sobre los protocolos y pruebas que se realizan; manipulación a través de un inventario automatizado, de los insumos que se consumen cada día; cálculos automatizados que ayudan a evitar el error humano; registro de muestras y de pruebas que se realicen; control de usuarios y monitoreo de las muestras en cada fase de análisis.

CAPÍTULO VI. CONCLUSIONES Y RECOMENDACIONES

VI.1 Conclusiones

De acuerdo a los resultados obtenidos en el desarrollo de este Trabajo Instrumental de Grado, se concluye lo siguiente:

- La implementación del sistema permitió al equipo de Biólogos e investigadores de la Fundación Jacinto Convit eliminar la inversión de tiempo que dedicaban a la búsqueda de información en diversas fuentes de conocimiento para la aplicación de pruebas, el registro manual de los insumos que se usarían durante las pruebas, así como los cálculos manuales para la ejecución de un análisis.
- La implementación de un inventario que puede disminuir los recursos usados automáticamente, permitió que el equipo del laboratorio pudiera enfocar sus esfuerzos en las ejecuciones de las pruebas y en el diagnóstico al que llegarán para cada uno, además de mantener la información del inventario actualizada.
- Dividir el proyecto en dos capas principales como *Back-End* y *Front-End* permitió un manejo considerablemente sencillo que permitía trabajar las interfaces del cliente independientemente de los procesos que se desarrollar en la capa del servidor.

- La aplicación del *framework* Spring Boot permitió que el desarrollo del sistema fuera más sencillo y dejando la codificación del sistema simple de entender y mantener.
- El uso de *Crystal Clear* como marco metodológico, permitió promover la comunicación con los clientes que a su vez también fueron los usuarios finales y la realización de continuas entregas que permitiera obtener un *feedback*.
- El uso del *framework* Angular para el *Front-end* facilitó la codificación, la organización del código a través de la creación de componentes, el uso de sus directivas para el manejo de la data desde el *template*. Además del fácil manejo para pasar de una vista a otra.
- La implementación de un módulo de registro de procesamiento de pruebas permitió tener un control y monitoreo del paso de las muestras por la fases de análisis

VI.2 Recomendaciones

- Se recomienda agregar un módulo que permita generar archivos .pdf en los cuales se pueda resumir todo el proceso de pruebas de una muestra, de esta forma se podrá compartir resultados con otras instituciones u hospitales que puedan aportar un diagnóstico que quizás no se haya tomado en cuenta desde la Fundación.
- Agregar un módulo que complemente al inventario en el cual se pueda mapear las estructuras donde se encuentran los reactivos o productos, no solo indicando el lugar donde están alojados sino más específicamente la sección de su ubicación.
- Ampliar el módulo de notificaciones de fallos, es recomendable la creación de una opción que permita realizar los pedidos directamente al proveedor o la casa comercial del insumo que se encuentre en menor cantidad dentro del inventario.
- Agregar opciones en el módulo de *wiki* que permitan añadir información como tablas e imágenes de forma intuitiva para los usuarios que manipulan este módulo.

Referencias Bibliográficas

[1] ABC, Diccionario. (2017). *Definición de Análisis*. Recuperado el 15 de Marzo de 2017, de Definición ABC, Sitio web: http://www.definicionabc.com/ciencia/analisis.php

[2] Angular, Team. (s.f.). *Angular Architecture*. Recuperado el 20 de Enero de 2018, de Angular, sitio web: https://angular.io/guide/architecture

[3] Cockburn, A. (2004). *Crystal Clear: A Human-Powered Methodology for Small Teams: A Human-Powered Methodology for Small Teams*. Person Education.

[4] Diagnostics, R. M. (2016). *Roche Molecular Diagnostics*. Recuperado el 18 de Marzo de 2017, de Roche Molecular Diagnostics, Sitio Web: https://molecular.roche.com/innovation/pcr

[5] Edu 4 Java. (2017). *Que es un Contenedor de Servlets. Instalación Apache Tomcat*. 3 de Mayo de 2017, de Edu 4 Java, Sitio web: http://www.edu4java.com/es/servlet/servlet1.html

[6] Johnson, R., Hoeller, J., & Donald, K. (s.f.). *Spring*. Recuperado el Mayo de 2017, de Spring, Sitio Web: http://docs.spring.io/spring/docs/3.2.18.RELEASE/spring-framework-reference/htmlsingle/#overview

[7] Julia. (2017). *Julia*. 19 de Mayo de 2017, de Julia, Sitio web: https://julialang.org/

[8] JHipster (Octubre de 2016). *Technology Stack*. Recuperado el 25 de Febrero de 2017, de JHipster Tech Sitio web: https://www.jhipster.tech/tech-stack/

[9] Martinez, R. (01 de Octubre de 2010). *PostgreSQL*. Recuperado el 20 de Mayo de 2017, de PostgreSQL, Sitio Web: http://www.postgresql.org.es/sobre_postgresql

[10] Network, M. D. (16 de Agosto de 2016). *CSS*. Recuperado el 08 de Mayo de 2017, de Mozilla Developer Network, Sitio Web: https://developer.mozilla.org/es/docs/Web/CSS

[11] Network, M. D. (s.f.). *HTML5*. Recuperado el Mayo de 2017, de MDN, Sitio Web: https://developer.mozilla.org/es/docs/HTML/HTML5

[12] Octave. (2017). *GNU Octave*. 19 de Mayo de 2017, de GNU Octave Sitio web: https://www.gnu.org/software/octave/

[13] Perry, J. S. (03 de Diciembre de 2012). *Introducción a la programación Java*. Recuperado el 06 de Mayo de 2017, de IBM Developer Words, Sitio Web: https://www.ibm.com/developerworks/ssa/java/tutorials/j-introtojava1/

[14] Rubio, M. (16 de Novembre de 2010). *¿Cómo funciona el sistema web?*. Recuperado el 06 de Mayo de 2017, de Altenwald, Sitio Web: http://altenwald.org/2010/11/16/como-funciona-el-sistema-web/

[15] Ruiz Pacheco, J. C. (2017). *Inyección de dependencias*. Recuperado el 22 de Mayo de 2017, de Microsoft Developer Network, Sitio Web: https://msdn.microsoft.com/es-es/communitydocs/net-dev/csharp/inyeccion-de-dependencias

[16] Spring. (2014). *Spring boot*. Recuperado el Mayo de 2017, de Spring, Sitio Web: http://projects.spring.io/spring-boot/

[17] Spring. (2017). *Spring data JPA*. Recuperado el 15 de Mayo de 2017, de Spring, Sitio Web: http://projects.spring.io/spring-data-jpa/

[18] Spring. (2017). *Understanding REST*. Recuperado el 20 de Mayo de 2017, de Spring, Sitio Web: https://spring.io/understanding/REST

[19] Stack Overflow. (2016). *Developer Survey Results*. Recuperado en Febrero de 2017, de Stack Overflow Sitio web: https://insights.stackoverflow.com/survey/2016#technology

[20] Tiobe. (18 de Enero de 2017). *Top index for January 2017*, Recuperado en Febrero de 2017, de Tiobe Sitio web: https://www.tiobe.com/tiobe-index/

[21] Victor, A., & Soria, V. (s.f) *Crystal Clear Methodology*, Recuperado el 10 de Enero de 2018, de Scribd, Sitio web: https://es.scribd.com/doc/55555056/Crystal-Clear-Version-Open-Document

APÉNDICES

Apéndice A - Glosario

Apache	Es un servidor web HTTP de código abierto
Back-end	Hace referencia al lado del servidor, quien contiene la capa de manejo de datos
Feedback	Palabra del inglés que significa retroalimentación o dar respuesta a un pedido.
Framework	Es un entorno o marco de trabajo que contiene un conjunto de herramientas, librerías, conceptos y buenas practicas, que encapsulan tareas repetitivas.
Front-end	Hace referencia al lado del cliente, en el que se manejan las tecnologías que corren en el navegador web.
HTTP	Protocolo de comunicación, utilizado por defecto en la web para la transferencia de información.
Java Persistence API	Conocido como JPA, es el API de persistencia desarrollada para la plataforma Java. Maneja datos relaciones en aplicaciones.
Jetty	Es un servidor web y contenedor de Servlets que puede ser embebido en las aplicaciones
Perfomance	Hace referencia al desempeño con respecto al rendimiento de un dispositivo o sistema
Scaffolding	Hace referencia a los elementos que se generan automáticamente (pueden ser modelos, vistas, clases controladoras) para la creación de páginas web.
Servlet	Un contenedor servlet es un programa capaz de recibir peticiones de páginas web y redireccionarlas a un objeto Servlet

Software	Se refiere a la parte lógica de las computadoras , es decir, instrucciones y datos
Template	Un template o plantilla o dispositivo de interfaz que proporciona una separación entre la estructura y contenido.
Undertow	Es un servidor web escrito en Java
Volumen	Corresponde a la medida del espacio que ocupa un cuerpo, la unidad para la medida del volumen es el litro (L)

Apéndice B – Exámenes ofrecidos por la UDM

Enfermedades Infecciosas

- VIH
- Tuberculosis
- Perfil Meningitis
  - Meningitis bacteriana
    - *Streptococcus pneumoniae*
    - *Haemophilus influenzae* tipo b
    - *Neisseria meningitidis*
    - *Streptococcus* grupo B
    - *Listeria monocytogenes*
  - Meningitis viral
    - Cytomegalovirus (CMV)
    - Human herpes virus 6 (HHV6)
    - Epstein-Barr virus (EBV)
    - Herpes simplex virus 2 (HSV2)
    - Varicella-zoster virus (VZV)
    - Herpes simplex virus 1 (HSV1)
    - Enterovirus humano

Enfermedades neoplásicas

- Detección de Leucemias
  - Panel LMA
    - PML/RARα
    - FLT3/ITD
    - Inv. (16)
    - AML1/ETO
    - BCR/ABL
  - Panel LMC
    - BCR/ABL
  - Panel LLA
    - TEL/AML1
    - MLL/AF4
    - TCF3/PBX1
    - BCR/ABL

Apéndice C - Requerimientos funcionales e historias de usuario

- El sistema deberá permitir la creación de usuarios administradores.
- El sistema deberá proporcionar tres roles para la creación de usuarios: administrador, investigador y consultor.
- El usuario administrador deberá crear, modificar y desactivar usuarios, podrá asignarles permisología.
- El sistema deberá proporcionar un módulo tipo wiki que permita el ingreso, edición y consulta de información sobre pruebas que se realizan en la Fundación.
- El sistema deberá contener un módulo que permita realizar los cálculos necesarios
- El sistema deberá permitir la creación y/o agregación de reactivos en el módulo de inventario.
- El sistema deberá permitir la creación y/o agregación de productos en el módulo de inventario.
- El usuario con permisología deberá tener la posibilidad de disminuir un reactivo o consumible manualmente dentro del inventario del sistema.
- El sistema deberá tener la posibilidad de disminuir del inventario la cantidad de un reactivo usado en una prueba, durante la fase de análisis del procesamiento de muestras.
- El sistema deberá disminuir del inventario la cantidad de un producto usado en una prueba, durante la fase de análisis del procesamiento de muestras.

- El sistema deberá realizar cálculos que permitan hallar los volúmenes finales de los reactivos que lo ameriten.
- El sistema deberá proporcionar el módulo de pre-análisis que permita agregar una muestra.
- El sistema deberá proporcionar el módulo de Análisis en el cual se realizará el registro de proceso de una prueba que será aplicada a una muestra.
- El sistema deberá proporcionar el módulo de Post-Análisis en el cual se registrará el resultado y/o diagnóstico de la prueba.
- El sistema deberá permitir la configuración de las notificaciones en la cual se pueda establecer los correos de los usuarios que recibirán las notificaciones.
- El sistema deberá proporcionar un histórico en el que se visualicen las fases de pre-análisis, análisis y post-análisis y las muestras asociadas a estas.
- El sistema deberá proporcionar un módulo de estadísticas en el cual se pueda consultar la cantidad de pruebas que puedan realizarse con el contenido de un producto.
- El sistema deberá conectarse con el sistema de mantenimiento de la Fundación para informar los equipos registrados.

Historias de usuario.

Tabla 3 *Historia de usuario 001*

Historia de usuario - 001			
Nombre	Registrar usuario	**Actor**	Administrador
Descripción	Como administrador quiero poder registrar un usuario para que accedan al sistema		
Observaciones	-		
Criterios de aceptación	1. Se deberá ingresar un número de cédula válido que no se encuentre ya registrado en el sistema. 2. Se podrá registrar un correo electrónico válido que no esté registrado en el sistema. 3. Se podrá registrar un nombre de usuario que no esté almacenado en el sistema.		

Fuente: Elaboración propia

Tabla 4 *Historia de usuario 002*

Historia de usuario - 002			
Nombre	Autenticar usuario	**Actor**	Usuario del sistema
Descripción	Como personal de la fundación quiero poder iniciar sesión con mi correo y clave		
Observaciones	-		
Criterios de aceptación	1. Un usuario se puede autenticar en el sistema ingresado su correo y contraseña. 2. En caso de olvido de contraseña podrá recuperarla en la opción de recuperar contraseña		

Fuente: Elaboración propia

Tabla 5 *Historia de usuario 003*

Historia de usuario - 003			
Nombre	Recuperar contraseña	**Actor**	Usuario del sistema
Descripción	Como personal de la fundación quiero recuperar mi contraseña por olvido		
Observaciones	-		
Criterios de aceptación	1. Se podrá recuperar la contraseña ingresando el correo electrónico. 2. Se enviará al correo del usuario un link que caducará luego de veinticuatro horas. 3. Al acceder al link se mostrarán los campos para ingresar una nueva contraseña.		

Fuente: Elaboración propia

Tabla 6 *Historia de usuario 004*

Historia de usuario - 004			
Nombre	Modificar datos de usuario	**Actor**	Investigador/Consultor
Descripción	Como personal de la fundación quiero poder modificar algunos de mis datos en el sistema		
Observaciones	-		
Criterios de aceptación	1. Un usuario podrá modificar sus datos. 2. Solo un administrador podrá modificar los cargos de un usuario. 3. El nombre de usuario no podrá ser modificado. 4. La cédula del usuario no podrá ser modificada.		

Fuente: Elaboración propia

Tabla 7 *Historia de usuario 005*

Historia de usuario - 005			
Nombre	Desactivar/activar un usuario	**Actor**	Administrador
Descripción	Como administrador quiero poder desactivar o activar a un usuario		
Observaciones	-		
Criterios de aceptación	1. Un usuario podrá ser desactivado por el administrador desde la lista de usuarios.		

Fuente: Elaboración propia

Tabla 8 *Historia de usuario 006*

Historia de usuario - 006			
Nombre	Registrar casa comercial	**Actor**	Investigador
Descripción	Como investigador quiero poder registrar una casa comercial		
Observaciones	-		
Criterios de aceptación	1. Se deberá validar que el nombre de la casa comercial no se encuentre ya almacenado en el sistema.		

Fuente: Elaboración propia

Tabla 9 *Historia de usuario 007*

Historia de usuario – 007			
Nombre	Registrar tipo de producto	**Actor**	Investigador
Descripción	Como investigador quiero poder registrar un tipo de producto		
Observaciones	-		
Criterios de aceptación	1. Se deberá validar que el nombre del tipo de producto no se encuentre ya almacenado en el sistema.		

Fuente: Elaboración propia

Tabla 10 *Historia de usuario 008*

Historia de usuario - 008			
Nombre	Registrar unidad de medida	**Actor**	Investigador
Descripción	Como investigador quiero poder registrar una unidad de medida		
Observaciones	-		
Criterios de aceptación	1. Se deberá validar que el nombre de la unidad no se encuentre ya almacenada en el sistema. 2. Se deberá clasificar en unidad de tipo inventario si son descriptivas para cantidades como por ejemplo unidades en caja o unidades en bolsa, de lo contrario la unidad será tipo simple.		

Fuente: Elaboración propia

Tabla 11 *Historia de usuario 009*

Historia de usuario - 009			
Nombre	Registrar un reactivo	**Actor**	Investigador
Descripción	Como investigador quiero poder registrar un reactivo		
Observaciones	-		
Criterios de aceptación	1. Deberá contener una opción que excluya a ciertos reactivos del registro de concentraciones. 2. Existirán algunos reactivos de los cuales no desea llevar su conteo en inventario.		

Fuente: Elaboración propia

Tabla 12 *Historia de usuario 010*

Historia de usuario - 010			
Nombre	Registrar un producto	**Actor**	Investigador
Descripción	Como investigador quiero poder registrar un producto		
Observaciones	-		
Criterios de aceptación	1. El registro de productos vinculará con casa comercial, tipos de productos y unidades.		

Fuente: Elaboración propia

Tabla 13 *Historia de usuario 011*

Historia de usuario - 011			
Nombre	Consultar productos en inventario	**Actor**	Investigador
Descripción	Como investigador quiero poder consultar los productos del inventario		
Observaciones	-		
Criterios de aceptación	1. Se deberá listar los productos registrados, mostrando la casa comercial a la que pertenecen, sus características y la cantidad en unidades. 2. Se podrá realizar la búsqueda de un producto por nombre, tipo y casa comercial		

Fuente: Elaboración propia

Tabla 14 *Historia de usuario 012*

Historia de usuario - 012			
Nombre	Consulta de reactivos en inventario	**Actor**	Investigador
Descripción	Como investigador quiero poder consultar los reactivos del inventario		
Observaciones	-		
Criterios de aceptación	1. Se deberá listar los reactivos registrados, mostrando la casa comercial a la que pertenecen, sus características, el lote y cantidades disponibles. 2. Se podrá realizar la búsqueda de un reactivo por nombre, casa comercial o siglas.		

Fuente: Elaboración propia

Tabla 15 *Historia de usuario 013*

Historia de usuario - 013			
Nombre	Crear una prueba	**Actor**	Investigador/Consultor
Descripción	Como usuario del sistema quiero poder crear pruebas en la sección wiki del sistema		
Observaciones	-		
Criterios de aceptación	1. Se deberán listar los productos registrados para su selección múltiple dentro de esta vista. 2. Se deberá listar los reactivos registrados para su selección simple o múltiple en esta vista. 3. Se podrán registrar las cantidades necesarias de cada uno de los productos o reactivos seleccionados, según la necesidad de la prueba. 4. Se podrá agregar información desde un editor de texto que permita manejar tamaños o colores de fuentes.		

Fuente: Elaboración propia

Tabla 16 *Historia de usuario 014*

Historia de usuario - 014			
Nombre	Modificación de pruebas	**Actor**	Investigador/Consultor
Descripción	Como usuario del sistema quiero poder editar la información de una prueba ya creada.		
Observaciones	-		
Criterios de aceptación	1. La edición de pruebas podrá realizarse para todos los datos de las pruebas, se podrán modificar las cantidades de los productos o los reactivos.		

Fuente: Elaboración propia

Tabla 17 *Historia de usuario 015*

Historia de usuario - 015			
Nombre	Cálculos de pruebas	**Actor**	Investigador
Descripción	Como investigador quiero poder realizar cálculos de una prueba sin que afectar el inventario		
Observaciones	-		
Criterios de aceptación	1. El manejo de esta opción podrá hacerse desde la sección wiki. 2. Los cálculos de volúmenes iniciales para PCR se realizaran con la fórmula C1 * V1 = C2 * V2 donde C representa las concentraciones iniciales y final respectivamente y V hace referencia al volumen. 3. Las concentraciones inicial y final deberán haber sido registradas en la misma unidad de medida en el registro del reactivo. 4. Existirán reactivos que no contienen concentraciones.		

Fuente: Elaboración propia

Tabla 18 *Historia de usuario 016*

Historia de usuario - 016			
Nombre	Registro de muestras	**Actor**	Investigador
Descripción	Como investigador quiero poder registrar una muestra en la fase de análisis		
Observaciones	-		
Criterios de aceptación	1. Debe guardarse la fecha y el usuario que registra la muestra.		

Fuente: Elaboración propia

Tabla 19 *Historia de usuario 017*

Historia de usuario - 017			
Nombre	Aplicar una prueba	**Actor**	Investigador
Descripción	Como investigador quiero poder aplicar una prueba a una muestra en la fase de análisis		
Observaciones	-		
Criterios de aceptación	1. Se podrá aplicar la prueba a una o más muestras registradas. 2. Por cada prueba aplicada debe agregarse un control positivo, un control negativo y uno reactivo. 3. Se desplegará una advertencia si la cantidad de productos o reactivos que hay en existencia del inventario, no es suficiente para aplicar la prueba. 4. El análisis se ejecutará si el usuario lo desea, Aun cuando no se cuente los insumos suficientes, en este caso no quedará registro la cantidad de productos y/o reactivos usados.		

Fuente: Elaboración propia

Tabla 20 *Historia de usuario 018*

Historia de usuario - 018			
Nombre	Registra un post-análisis	**Actor**	Investigador
Descripción	Como investigador quiero poder registrar los resultados de una prueba en la fase de post-análisis		
Observaciones	-		
Criterios de aceptación	1. Se deberá poder cargar una imagen que representa el resultado de la prueba realizada		

Fuente: Elaboración propia

Tabla 21 *Historia de usuario 019*

Historia de usuario - 019			
Nombre	Configuración de correos para notificaciones	**Actor**	Administrador
Descripción	Como investigador quiero poder agregar o eliminar correos electrónicos de la lista de usuarios que reciben notificaciones.		
Observaciones	-		
Criterios de aceptación	1. La configuración deberá mostrar una lista con los correos electrónicos de los usuarios registrados.		

Fuente: Elaboración propia

Tabla 22 *Historia de usuario 020*

Historia de usuario - 020			
Nombre	Búsqueda en histórico	**Actor**	Usuario del sistema
Descripción	Como usuario del sistema quiero poder realizar búsquedas en el histórico de muestras por rango de fechas		
Observaciones	-		
Criterios de aceptación	1. La visualización de muestras en el histórico deberá realizarse por fase. 2. Se podrá consultar información detalla de las muestras		

Fuente: Elaboración propia

Tabla 23 *Historia de usuario 021*

Historia de usuario - 021			
Nombre	Consultar estadísticas	**Actor**	Usuario del sistema
Descripción	Como usuario del sistema quiero poder visualizar las estadísticas disponibles en el sistema		
Observaciones	-		
Criterios de aceptación	1. Se mostrarán gráficos que ayuden a visualizar la información obtenida.		

Fuente: Elaboración propia

Apéndice D – Encuestas y evaluación de tecnologías

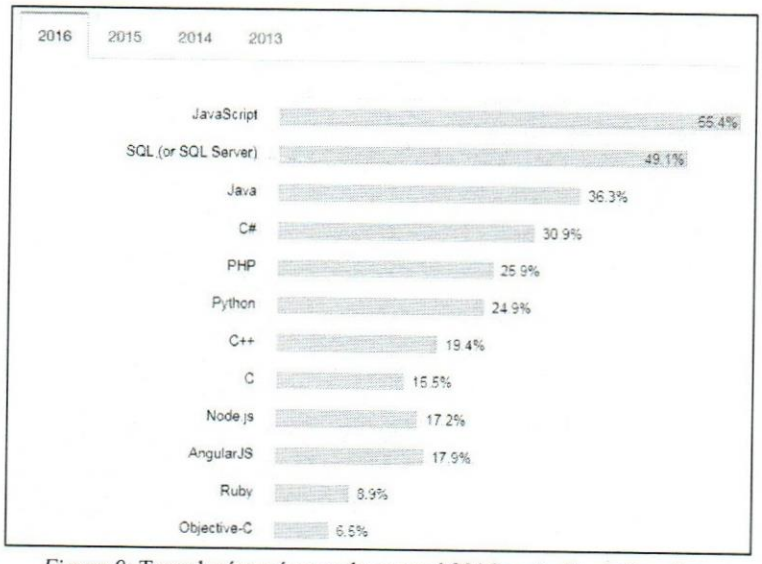


Figura 9: Tecnologías más populares en el 2016 según Stack Overflow
Fuente [19]

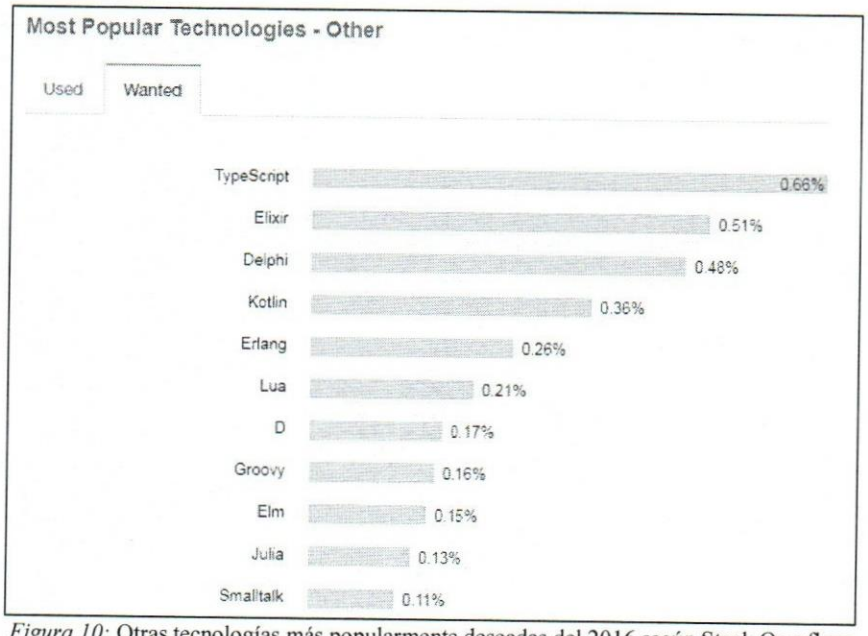
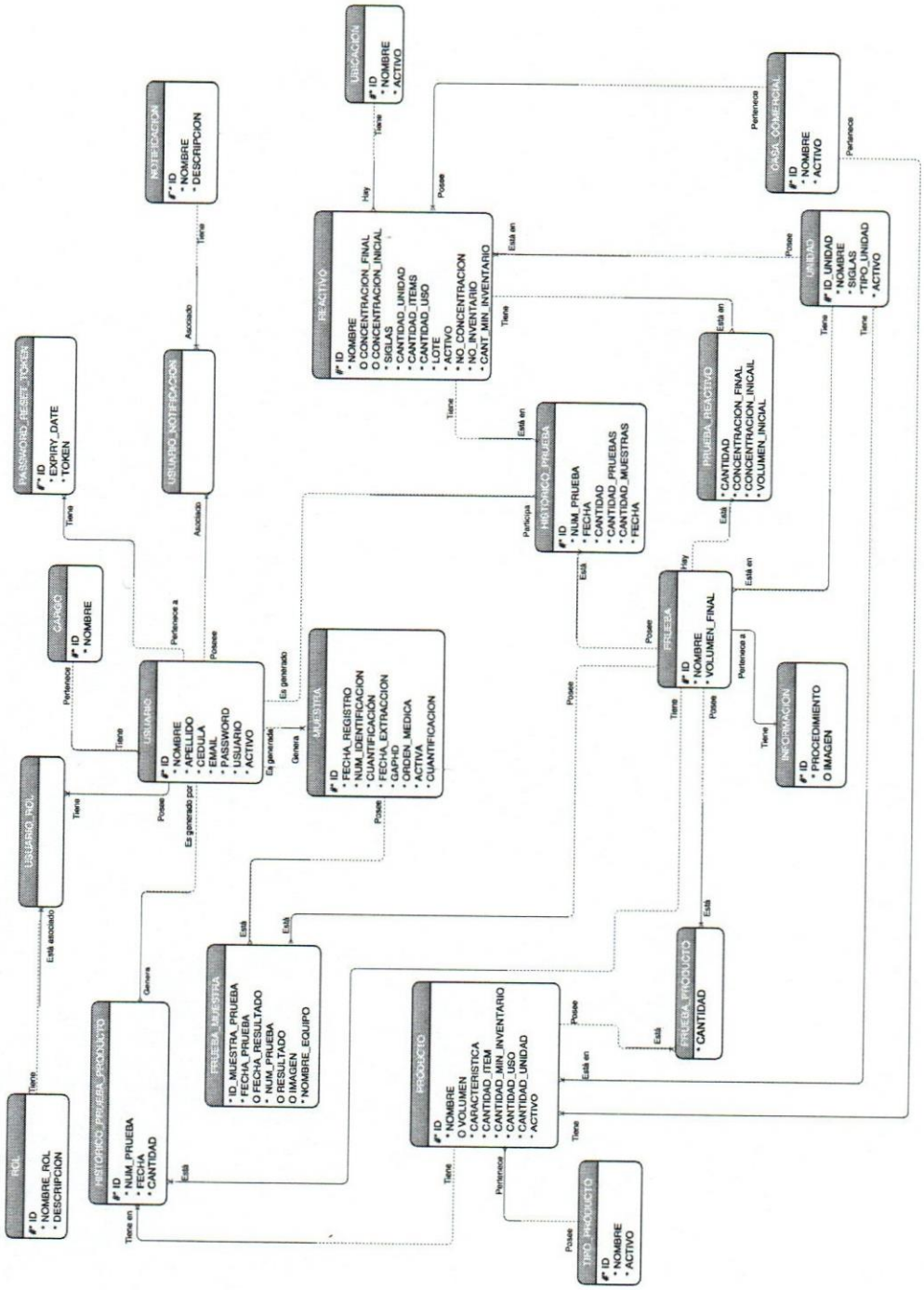


Figura 10: Otras tecnologías más popularmente deseadas del 2016 según Stack Overflow
Fuente [19]

Figura 11: Top de los lenguajes de programación según TIOBE
Fuente [20]

Apéndice E – Modelo Entidad - Relación



Apéndice F- Ilustraciones del sistema

F.1 Pantalla de registro de usuario

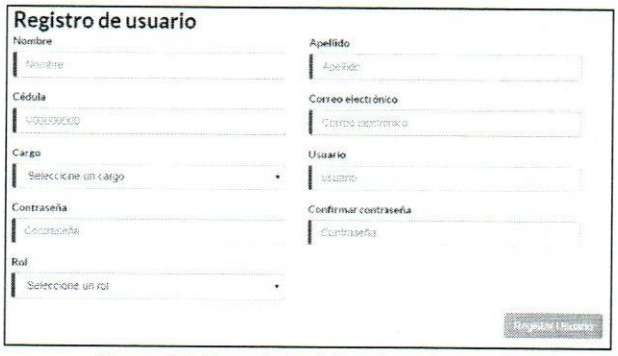


Figura 12: Formulario del registro de usuario
Fuente: Elaboración Propia

F.2 Pantalla de listado de usuarios y edición de datos

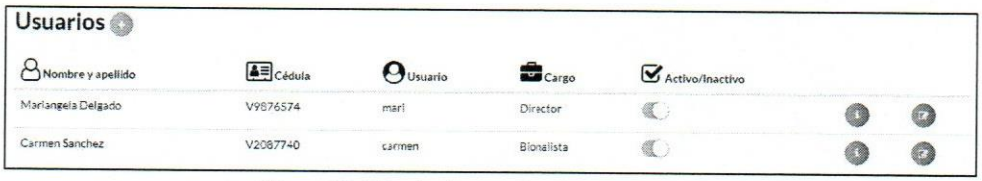


Figura 13: Listado de usuarios del sistema
Fuente: Elaboración propia

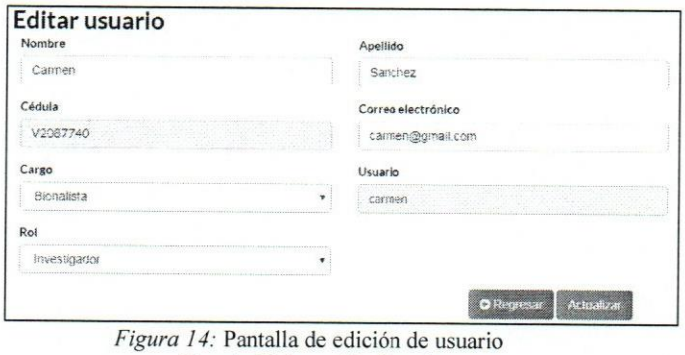


Figura 14: Pantalla de edición de usuario
Fuente Elaboración Propia

F.3 Pantalla inicio de sesión

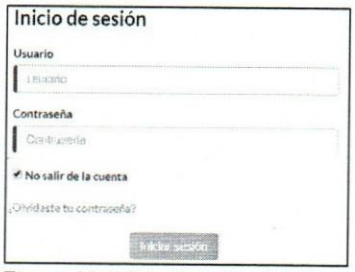


Figura 15: Inicio de sesión del sistema
Fuente: Elaboración propia

F.4 Pantallas para la recuperación de contraseña

Figura 17: Recuperación de contraseña
Fuente Elaboración propia

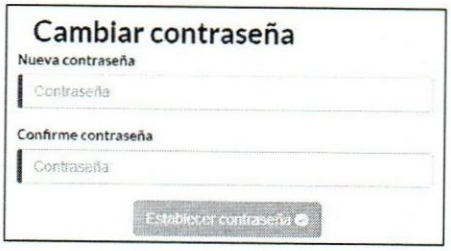


Figura 16 : Cambio de contraseña
Fuente: Elaboración propia

F.5 Pantallas de listado de casas comerciales y unidades de medida

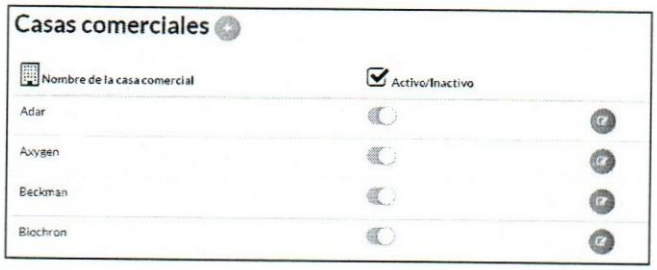
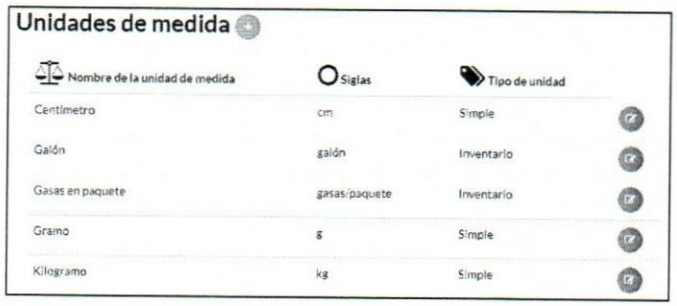


Figura 18: Listado de casas comercial
Fuente: Elaboración propia

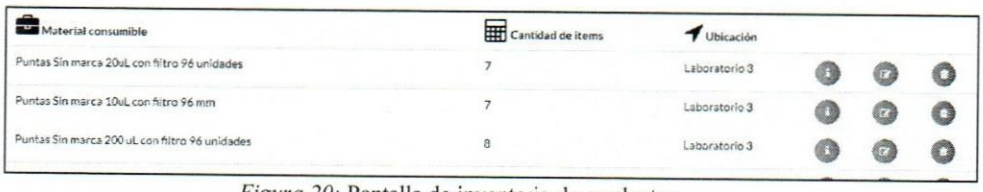


Unidades de medida

Nombre de la unidad de medida	Siglas	Tipo de unidad
Centímetro	cm	Simple
Galón	galón	Inventario
Gasas en paquete	gasas/paquete	Inventario
Gramo	g	Simple
Kilogramo	kg	Simple

Figura 19: Listado de unidades de medida
Fuente: Elaboración propia

F.6 Pantalla del inventario de productos



Material consumible	Cantidad de items	Ubicación
Puntas Sin marca 20uL con filtro 96 unidades	7	Laboratorio 3
Puntas Sin marca 10uL con filtro 96 mm	7	Laboratorio 3
Puntas Sin marca 200 uL con filtro 96 unidades	8	Laboratorio 3

Figura 20: Pantalla de inventario de productos
Fuente: Elaboración propia

F.7 Pantalla de registro de productos

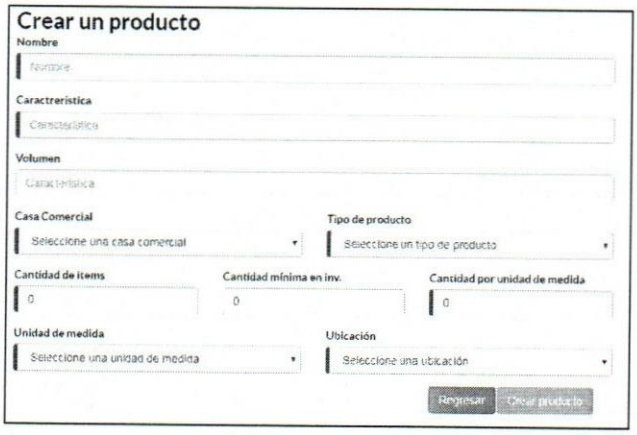


Figura 21: Creación de un producto
Fuente: Elaboración propia

F.8 Pantalla de registro de reactivos

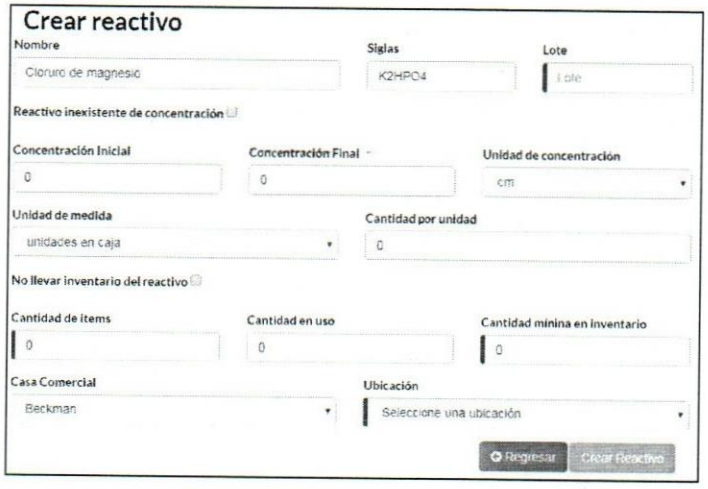


Figura 22: Creación de un reactivo
Fuente: Elaboración propia

F.9 Creación de pruebas en la wiki

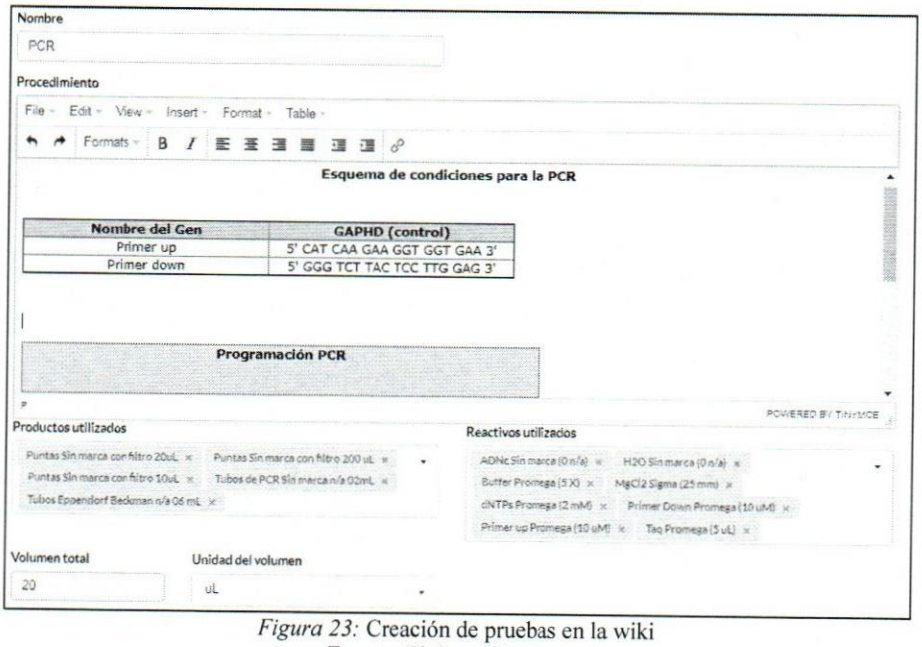


Figura 23: Creación de pruebas en la wiki
Fuente: Elaboración propia

F.10 Pantalla del módulo de cálculo por prueba

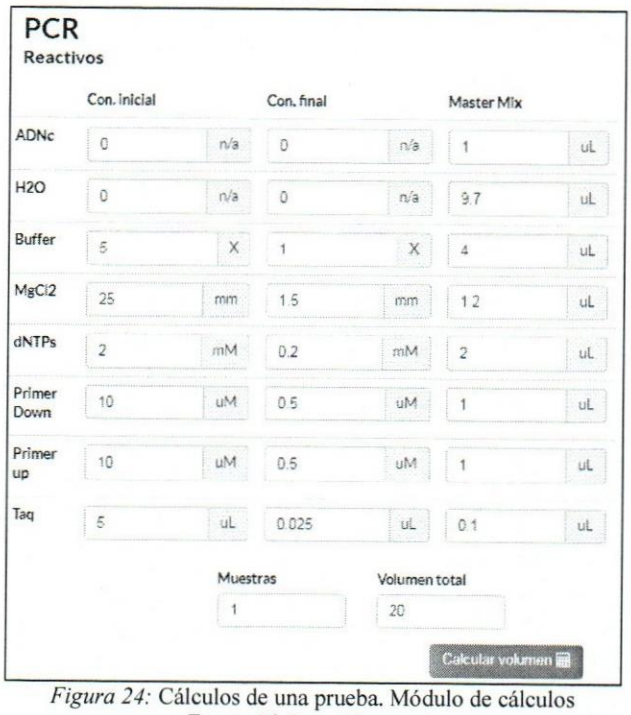


Figura 24: Cálculos de una prueba. Módulo de cálculos
Fuente Elaboración propia

F.11 Pantalla de la fase de Pre-Análisis

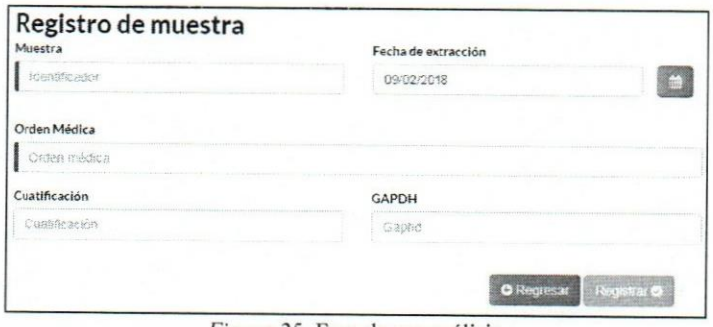


Figura 25: Fase de pre-análisis
Fuente: Elaboración propia

F.12 Pantalla de la fase de análisis

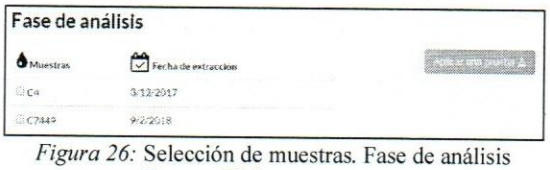


Figura 26: Selección de muestras. Fase de análisis
Fuente: Elaboración propia

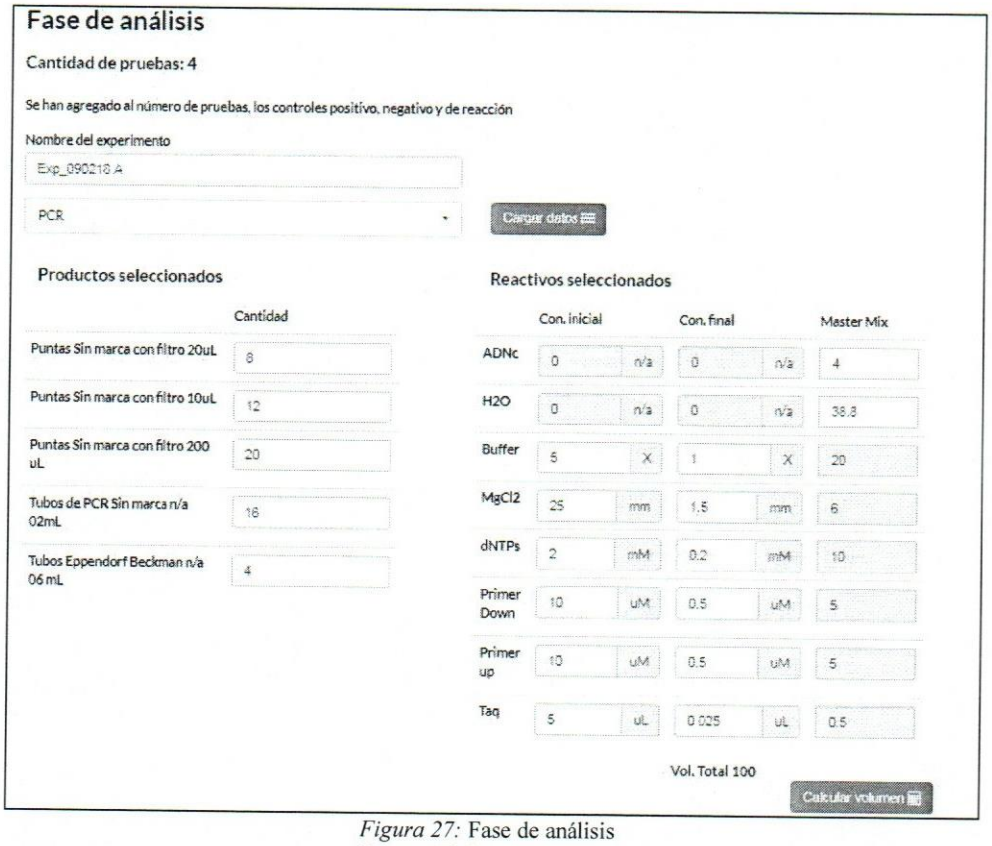


Figura 27: Fase de análisis
Fuente: Elaboración propia

F.13 Pantalla de la fase de Post-análisis

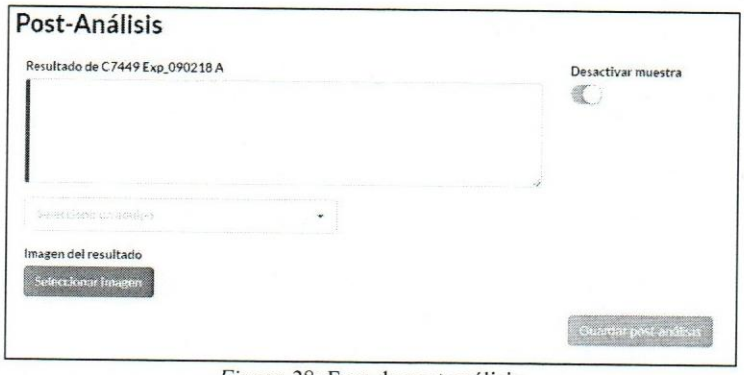


Figura 28: Fase de post-análisis
Fuente: Elaboración propia

F.14 Configuración de las notificaciones

Figura 29: Configuración de notificaciones
Fuente: Elaboración propia

F.15 Pantalla del histórico de muestras y sus fases

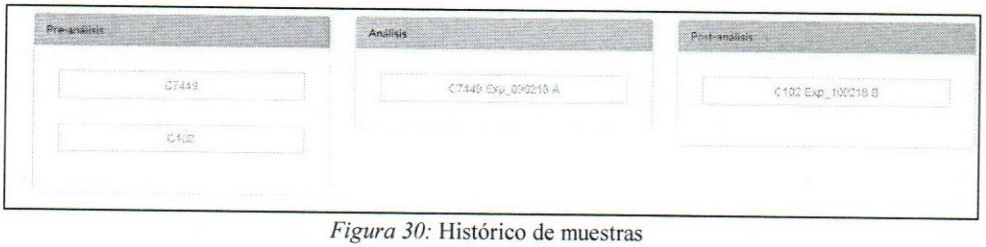


Figura 30: Histórico de muestras
Fuente: Elaboración propia

F.16. Pantalla detalle de la información de una muestra

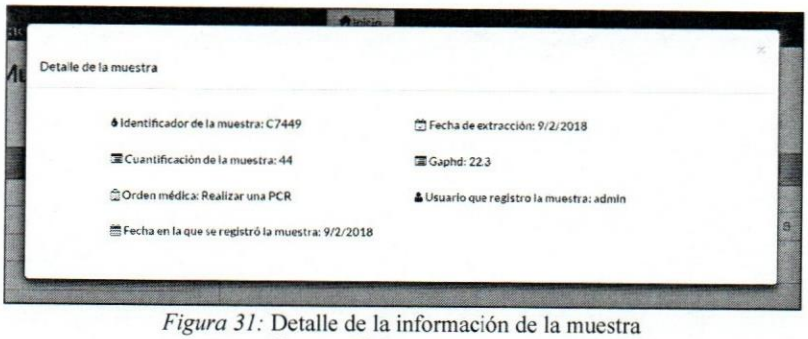


Figura 31: Detalle de la información de la muestra
Fuente: Elaboración propia

F.17 Estadísticas

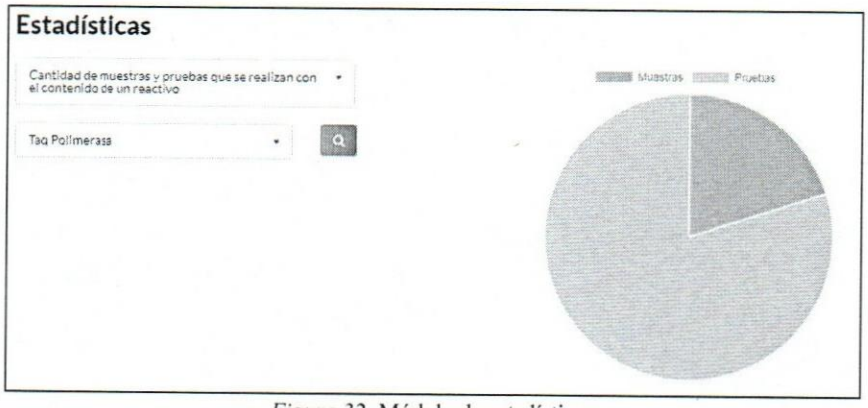


Figura 32: Módulo de estadísticas
Fuente: Elaboración propia

Apéndice G- Esquema de condiciones para la PCR

	Master mix
	1X
Buffer 5X	4 uL
MgCl2 (25 mM)	1,6 uL
dNTP's (2 mM)	2 uL
Primer up (10 uM)	1,5 uL
Primer down (10 uM)	1,5 uL
Taq	0,1 uL
H2O	8,3 uL
cDNA	1 uL
Vol. Total	20 uL

Figura 33 Master mix de una PCR
Fuente: Documentos de la Fundación Jacinto Convit

Programación PCR			
Etapas	**Temperatura (ºC)**	**Tiempo (min)**	**Ciclos**
Desnaturalización	95	4	1x
Desnaturalización	95	0,5	35x
Hibridación	50	1	
Extensión	72	1	
Extensión Final	72	7	1x

Tamaño producto		% de Gel de Agarosa
Set	Bp	1,5%
ADNc	243	TAE 1X
ADNg	347	

Figura 34 Programación para PCR
Fuente: Documentos de la Fundación Jacinto Convit

Apéndice F- Fórmula para el cálculo del volumen

$$C_1 * V_1 = C_2 * V_2$$

Donde: C_1 es la concentración inicial.

V_1 es el volumen inicial.

C_2 es la concentración final.

V_2 es el volumen final.

- La concentración inicial de un reactivo viene dada por la casa comercial a la que pertenezca.

- Las concentraciones finales son establecidas por el laboratorio para un experimento, suele haber un entandar establecido pero la decisión es de quien realiza el experimento.

- El volumen final lo determina quien hace la reacción, esto dependerá de la cantidad del experimento, representa el volumen total de reacción.

- El volumen inicial representa la cantidad necesaria del reactivo en el experimento.

Al requerir el valor de V_1 la fórmula que se está aplicando en el sistema es: $V_1 = \frac{C_2 \times V_2}{C_1}$

A continuación se presenta a modo general y de ejemplo el código que se aplicaría para del cálculo del volumen inicial de un solo reactivo:

```
if ( !reagent.no_concentration ) {
  const finalConcen  = +this.finalCon;
  const finalVol     = this.proof.FinalVolume;
  const initialConce = +this.initialCon;
  let initialVol     = (finalConcen * finalVol)/initialConce;
  initialVol = initialVol * this.quantitySamples;
  this.initialVolume = initialVol;

} else if (this.firstCalculate) { //reactivos sin concentracion

    initialVol = this.initialVolume;
    initialVol = initialVol * this.quantitySamples;
    this.initialVolume = initialVol;
}

this.finalVolume = this.proof.FinalVolume * this.quantitySamples;
this.firstCalculate = false;
```

Figura 35: Ejemplo del código aplicado al cálculo del volumen de un reactivo
Fuente: Elaboración propia