

FACULTAD DE INGENIERÍA

Escuela Académico Profesional de Ingeniería de Sistemas e
Informática

Trabajo de Suficiencia Profesional

**Conocimiento aplicado como analista de
control de calidad de sistemas**

Brigham Sebastian Silva Moran

Para optar el Título Profesional de
Ingeniero de Sistemas e Informática

Huancayo, 2025

Repositorio Institucional Continental
Trabajo de suficiencia profesional digital



Esta obra está bajo una Licencia "Creative Commons Atribución 4.0 Internacional" .

INFORME DE CONFORMIDAD DE ORIGINALIDAD DE TRABAJO DE INVESTIGACIÓN



INFORME DE CONFORMIDAD DE ORIGINALIDAD DE TRABAJO DE INVESTIGACIÓN

A : Decano de la Facultad de Ingeniería
DE : DR. MAGLIONI ARANA CAPARACHIN
Asesor de trabajo de investigación
ASUNTO : Remito resultado de evaluación de originalidad de trabajo de investigación
FECHA : 17 de diciembre del 2025

Con sumo agrado me dirijo a vuestro despacho para informar que, en mi condición de asesor del trabajo de investigación:

Título:

Conocimiento Aplicado como Analista de Control de Calidad de Sistemas

Autor:

Brigham Sebastian Silva Moran – Carrera profesional Ingeniería de Sistemas e Informática

Se procedió con la carga del documento a la plataforma "Turnitin" y se realizó la verificación completa de las coincidencias resaltadas por el software dando por resultado 12 % de similitud sin encontrarse hallazgos relacionados a plagio. Se utilizaron los siguientes filtros:

- Filtro de exclusión de bibliografía SI NO
- Filtro de exclusión de grupos de palabras menores SI NO
Nº de palabras excluidas (en caso de elegir "SI"): 10.
- Exclusión de fuente por trabajo anterior del mismo estudiante SI NO

En consecuencia, se determina que el trabajo de investigación constituye un documento original al presentar similitud de otros autores (citas) por debajo del porcentaje establecido por la Universidad Continental.

Recae toda responsabilidad del contenido del trabajo de investigación sobre el autor y asesor, en concordancia a los principios expresados en el Reglamento del Registro Nacional de Trabajos conducentes a Grados y Títulos – RENATI y en la normativa de la Universidad Continental.

DEDICATORIA

Dedico este trabajo con profundo cariño a mi esposa y a mi hija, quienes han sido el pilar fundamental a lo largo de este camino. A mis padres, por su esfuerzo constante, por enseñarme el valor del compromiso y la perseverancia, y por brindarme su apoyo incondicional en cada etapa de mi formación profesional. A mis hermanos, por ser siempre fuente de motivación y fortaleza en los momentos difíciles. Esta meta alcanzada es tanto mía como de ustedes.

También dedico este logro a mí mismo, por no rendirme ante los desafíos y por seguir creyendo en la importancia de alcanzar los objetivos con esfuerzo, constancia y honestidad.

AGRADECIMIENTO

Agradezco, en primer lugar, a Dios por haberme dado la salud, la claridad y la fortaleza necesarias para culminar esta etapa académica y profesional.

A la Universidad Continental, por brindarme la oportunidad de formarme como profesional en la carrera de Ingeniería de Sistemas e Informática, y a cada uno de los docentes que, con su dedicación y exigencia, contribuyeron a construir las bases del conocimiento que hoy pongo en práctica.

Extiendo un especial agradecimiento a la institución Caja Huancayo por permitirme desarrollarme profesionalmente como analista de Control de Calidad de Software y por confiar en mis capacidades. A mis compañeros de trabajo, gracias por su apoyo, su guía y por permitirme ser parte de un equipo comprometido con la mejora continua y la calidad del servicio.

Finalmente, agradezco a familia que, de una u otra forma, me brindaron su ánimo, compañía y comprensión durante este proceso. Cada palabra de aliento y cada gesto de apoyo fueron fundamentales para alcanzar esta meta.

RESUMEN EJECUTIVO

Este trabajo de suficiencia profesional se enfoca en analizar el papel del analista de Control de Calidad de Software en la industria del desarrollo de *software*. El análisis se basa en la experiencia laboral propia del autor de este trabajo sobre buenas prácticas en el control de calidad, así como en el conocimiento obtenido en el campo a través del estudio.

El estudio comienza con una revisión a las funciones del analista de calidad de *software* y la empresa en la que labora con el fin de tener un contexto sobre el tipo de sistemas que son afectados y la minuciosidad con la que deben ser analizados. Se examinan sus responsabilidades específicas, incluyendo la creación de casos de prueba, ejecución de pruebas, reporte de defectos y colaboración con los equipos de funcionales y de desarrollo.

Lo expuesto revela lo fundamental que es el rol del analista de Control de Calidad en el ciclo de vida del *software* para garantizar la entrega de un producto de alta calidad. Se destaca la importancia de la colaboración entre los equipos de control de calidad y desarrollo para identificar y resolver problemas de manera ágil y efectiva.

Finalmente, este trabajo concluye con recomendaciones para mejorar la eficacia del proceso del Control de Calidad de Software como la adopción de metodologías ágiles, la comunicación interdepartamental y la importancia de estándares.

Palabras clave: analista de Control de Calidad de Software, pruebas de *software*, funciones, responsabilidades, metodologías ágiles

ABSTRACT

This professional sufficiency work focuses on analyzing the role of the software quality control analyst in the software development industry. The analysis is based on the author's own work experience regarding best practices in quality control, as well as knowledge obtained in the field through study.

The study begins with a review of the functions of the software quality analyst and the company they work for, to provide context about the types of systems affected and the thoroughness with which they must be analyzed. Their specific responsibilities are examined, including test case creation, test execution, defect reporting, and collaboration with functional and development teams.

The findings reveal the fundamental role of quality control analyst in the software lifecycle to ensure the delivery of a high-quality product. The importance of collaboration between quality control and development teams is emphasized to identify and resolve issues in an agile and effective manner.

Finally, this work concludes with recommendations to improve the effectiveness of the software quality control process, such as adopting agile methodologies, interdepartmental communication, and the importance of standards.

Keywords: software quality control analyst, software testing, functions, responsibilities, agile methodologies.

ÍNDICE

DEDICATORIA.....	4
AGRADECIMIENTO.....	5
RESUMEN EJECUTIVO	6
ABSTRACT	7
ÍNDICE	8
TABLA DE FIGURAS	11
INTRODUCCIÓN	12
1. CAPÍTULO I ASPECTOS GENERALES DE LA EMPRESA O INSTITUCIÓN.....	13
1.1. Datos Generales de la Institución.....	13
1.2. Actividades Principales de la Institución o Empresa	13
1.3. Reseña Histórica de la Institución o Empresa.....	14
1.4. Organigrama de la Institución o Empresa.....	14
1.5. Visión y Misión.....	15
1.6. Descripción del Área donde se Realizan las Labores Profesionales	15
1.7. Descripción del Cargo y Responsabilidades del Bachiller en la Empresa	15
2. CAPÍTULO II INFORMACIÓN GENERAL SOBRE LAS ACTIVIDADES PROFESIONALES	19
2.1. Diagnóstico Situacional.....	19
2.2. Detección de Oportunidades o Necesidades en el Campo Profesional	20
2.2.1. Metodologías ágiles.....	20
2.2.3. Responsables de monitoreo posproducción.....	23
2.3. Planteamiento del Problema.....	24
2.4. Objetivos de la Actividad Profesional	24
2.5. Justificación de la Actividad Profesional	25
2.6. Tipo y Enfoque de Investigación.....	25
2.7. Resultados Esperados	26
3. CAPÍTULO III MARCO TEÓRICO	27

3.1. Fundamentos Teóricos de las Actividades Desarrolladas	27
3.1.1. Proceso de verificación de <i>software</i> en la entidad financiera Caja Huancayo	27
3.1.2. Proceso de ejecución de pruebas de <i>software</i>	31
3.1.3. Proceso fundamental del <i>testing</i>	31
3.1.4. Validación de programas informáticos.....	32
3.1.5. Propósitos de la evaluación del <i>software</i>	33
3.1.6. Enfoque de las pruebas de <i>software</i>	34
3.1.7. Técnicas aplicadas al diseño de casos de pruebas	34
3.1.8. <i>Clasificación de pruebas</i>	36
3.1.9. Niveles en la ejecución de pruebas.....	37
4. CAPÍTULO IV DESCRIPCIÓN DE LAS ACTIVIDADES PROFESIONALES.....	38
4.1. Descripción de Actividades Profesionales	41
4.1.1. Enfoque de las actividades profesionales	41
4.1.2. Alcance de las actividades profesionales.....	41
4.1.3. Entregables de las actividades profesionales.....	42
4.2. Aspectos Técnicos de la Actividad Profesional	42
4.2.1. Metodologías.....	42
4.2.2. Técnicas.....	43
4.2.3. Instrumentos	44
4.2.4. Equipos y materiales utilizados en el desarrollo de las actividades	49
4.3. Ejecución de las Actividades Profesionales	49
4.3.1. Cronograma de actividades realizadas	49
4.3.2. Proceso y secuencia operativa de las actividades profesionales.....	51
5. CAPÍTULO V RESULTADOS	53
5.1. Resultados Finales de las Actividades Realizadas	53
5.2. Logros Alcanzados	54
5.3. Dificultades Encontradas.....	55
5.4. Planteamiento de Mejoras	55
5.4.1. Metodologías propuestas.....	56

5.4.2. Descripción de la implementación	56
5.5. Análisis.....	57
5.6. Aporte del Bachiller en la Empresa y/o Institución.....	57
CONCLUSIONES	59
RECOMENDACIONES	61
BIBLIOGRAFÍA.....	63
ANEXOS.....	65
Anexo 1. Fotografías	65

TABLA DE FIGURAS

Figura 1. Estructura organizacional de la entidad financiera Caja Huancayo.....	14
Figura 2. Ciclo de Vida del Desarrollo de Software	16
Figura 3. Análisis FODA del Dpto. de Sistemas de la Caja Huancayo.....	19
Figura 4. Proceso de Pruebas de Software en 3 capas.....	28
Figura 5. Arquitectura general del proyecto.....	39
Figura 6. Casos de uso del proyecto.....	40
Figura 7. Plan de pruebas del proyecto plataforma Web Cuenta Futuro.....	49
Figura 8. Cronograma de actividades profesionales.....	50

INTRODUCCIÓN

El presente estudio tiene como propósito recopilar, describir y analizar la aplicación de conocimientos obtenidos por el estudio y la práctica dentro de la Universidad Continental para la carrera de Ingeniería de Sistemas e Informática dentro de la experiencia laboral del autor de este informe y demostrar de esta manera que el autor se encuentra en la capacidad suficiente de ejercer y aplicar los conocimientos de esta carrera profesional como lo ha venido desempeñando dentro de la institución donde labora actualmente, la Caja Municipal de Ahorro y Crédito de Huancayo, dentro del área de sistemas e innovación. Esto con el fin de obtener el título profesional que corresponde a la escuela académico profesional de la facultad de ingeniería.

1. CAPÍTULO I

ASPECTOS GENERALES DE LA EMPRESA O INSTITUCIÓN

1.1. Datos Generales de la Institución

La Caja Municipal de Ahorro y Crédito de Huancayo, comúnmente conocida como Caja Huancayo, ha surgido como una notable institución financiera en el panorama peruano, destacando por su enfoque en el impacto social y su contribución al desarrollo económico local. Fundada en 1986, la Caja Huancayo ha desempeñado un papel crucial en la provisión de servicios financieros a microempresarios, agricultores y familias de bajos ingresos en la región central del país, con una sólida presencia en la región de Junín y diversas zonas del país (1).

La relevancia de la Caja Huancayo radica en su compromiso con la inclusión financiera y el empoderamiento económico de comunidades desfavorecidas. A través de una amplia gama de productos y servicios, que incluyen microcréditos, cuentas de ahorro y programas de educación financiera, entre otros. Esta institución ha logrado impulsar la actividad económica local y mejorar las condiciones de vida de miles de peruanos (1).

Uno de los aspectos más destacados de la Caja Huancayo es su enfoque en la responsabilidad social corporativa. La institución ha desarrollado programas de desarrollo comunitario, enfocados en áreas como la educación, la salud y el medio ambiente, con el objetivo de generar un impacto positivo en la sociedad y promover la sostenibilidad a largo plazo.

En conclusión, la Caja Huancayo se erige como un ejemplo inspirador de una institución financiera que va más allá de la maximización de beneficios, priorizando el bienestar de las comunidades a las que sirve. Su modelo de negocio centrado en el impacto social ofrece lecciones valiosas para el sector financiero y el desarrollo económico en general en Perú y más allá (2).

1.2. Actividades Principales de la Institución o Empresa

Las principales funciones de la Caja Municipal de Ahorro y Crédito de Huancayo son la prestación de servicios financieros a microempresarios, agricultores y familias, inicialmente en la región central del país. Sin embargo, a medida que la institución ha crecido, sus servicios se han extendido por todas las regiones del país, contando con más de 200 agencias a nivel nacional (2).

La Caja Huancayo se centra en la concesión de microcréditos, cuentas de ahorro y programas de educación financiera para promover el desarrollo económico local y la inclusión financiera. Además, la institución organiza actividades de responsabilidad social corporativa, como programas de desarrollo comunitario que abarcan áreas como la educación, la salud y el medio ambiente. Estas iniciativas buscan generar un impacto positivo en la sociedad y promover la sostenibilidad a largo plazo en las comunidades donde opera la organización (2).

1.3. Reseña Histórica de la Institución o Empresa

La Caja Municipal de Ahorro y Crédito de Huancayo fue fundada en el año 1986 en la ciudad de Huancayo, Perú. Surgió como una iniciativa para promover el desarrollo económico local y promocionar servicios financieros accesibles a la población de la región central del país. Desde sus inicios, la Caja Huancayo se ha destacado por el apoyo a familias de bajos recursos, trabajadores dedicados a la agricultura y microempresarios (1).

A lo largo de su historia, la Caja Huancayo ha experimentado un crecimiento significativo, expandiendo su presencia a través de una red de sucursales en diversas ciudades y pueblos en diferentes regiones del país. Esta expansión ha permitido que la institución llegue a un mayor número de clientes y amplíe su impacto en las comunidades donde opera (1).

1.4. Organigrama de la Institución o Empresa

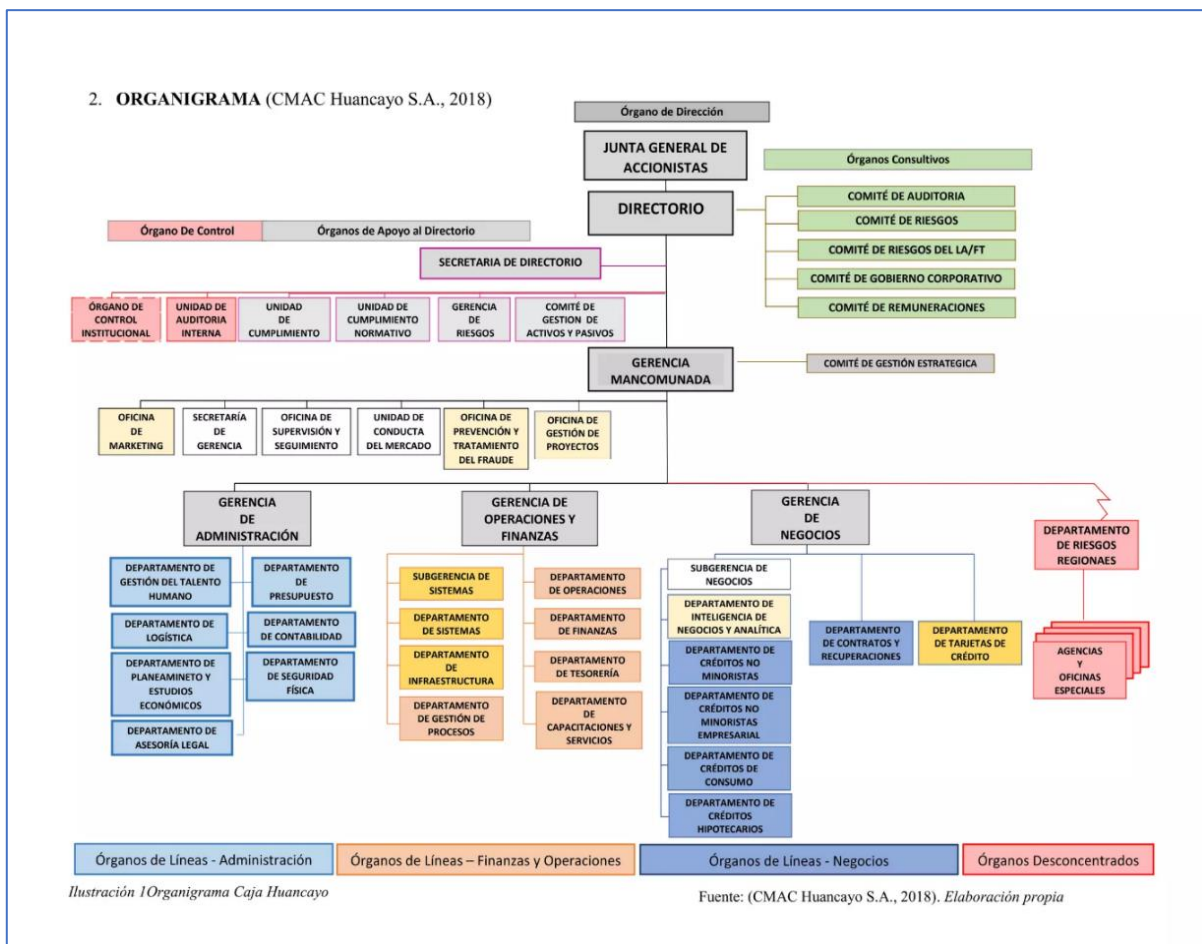


Figura 1. Estructura organizacional de la entidad financiera Caja Huancayo

Fuente: LINARES, Mitzi. Informe de Procesos de Caja Huancayo y Propuesta de Mejora. Perú [en línea]. [Fecha de consulta: 10 enero 2025]. (3) Disponible en: <https://es.slideshare.net/mlinaresvizcarra/informe-caja-huancayo>

1.5. Visión y Misión

Visión

La aspiración de la institución es ser el referente regional que posea una visión global en lo correspondiente al área financiera, ofreciendo servicios financieros que se consideren íntegros de clase mundial, enfocándose primordialmente en la satisfacción de sus usuarios quienes puedan verlos como un aliado de confianza (4).

Misión

El propósito de la institución es cambiar la vida de sus usuarios siendo un compañero y apoyo para alcanzar sus sueños durante cada etapa de sus vidas y haciendo de este mundo un lugar más sostenible (4).

1.6. Descripción del Área donde se Realizan las Labores Profesionales

El departamento de Sistemas e Innovación, que forma parte de la Caja Huancayo, atiende las solicitudes realizadas por las diferentes áreas y departamentos de la institución, según sus necesidades. Propone ideas innovadoras para agilizar los procesos internos, con el fin de sobresalir en el ámbito financiero. Además, soluciona incidentes relacionados con los sistemas implementados por los colaboradores y brinda mantenimiento para evitar desfasajes y obsolescencia.

Para ello, los colaboradores del departamento de sistemas están organizados en diferentes equipos de acuerdo al departamento que atienden y los módulos que éstos manejan, entre ellos están el módulo de ahorros, el módulo de créditos, el módulo de seguros, el módulo de canales electrónicos, entre otros. Cada equipo está conformado por analistas funcionales, analistas de desarrollo, analistas de Control de Calidad de Software y un subjefe de Desarrollo. Asimismo, todos los equipos del departamento están bajo la supervisión de un jefe del Departamento de Sistemas, quien está encargado de gestionar y mantener la infraestructura tecnológica que sustenta las operaciones financieras y comerciales de la organización (3).

El desarrollo de *software*, a diferencia de otros productos, es un proceso creativo. Para evitar que este proceso sea descontrolado, es importante establecer parámetros basados en técnicas derivadas tanto de la experiencia como de normas internacionales de estándares, con el fin de reducir el riesgo de fracaso (5).

1.7. Descripción del Cargo y Responsabilidades del Bachiller en la Empresa

Dentro del ciclo de vida del *software*, también conocido como SDLC, se describen diferentes pasos como parte de este proceso. Entre estos pasos se encuentran los procesos de planificación, desarrollo, prueba e implementación, estos siguen una secuencia donde cada etapa emplea el resultado de su fase anterior (6). La siguiente imagen ilustra el ciclo de vida de *software*:

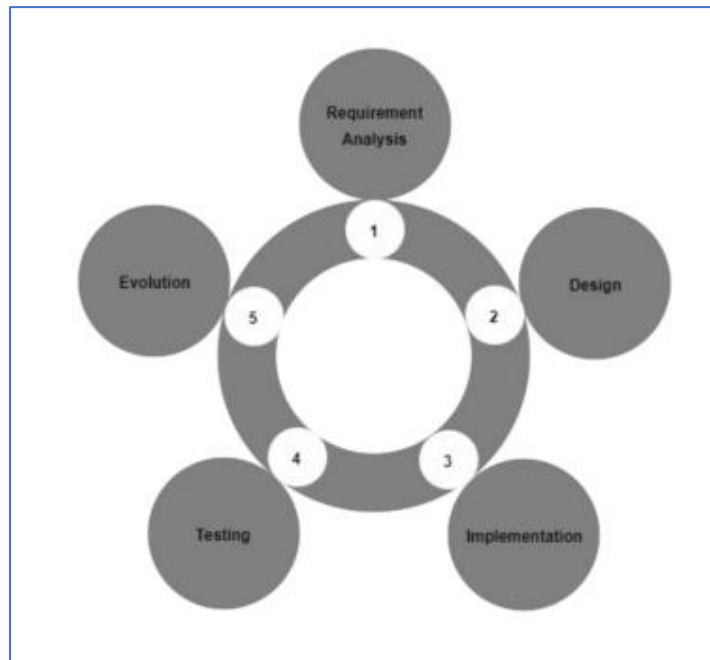


Figura 2. Ciclo de Vida del Desarrollo de Software

Fuente: Duquino, Ángela. Automatización de un Sistema de Pruebas de Software para la Optimización del Proceso de Calidad de DetectID. Bogotá, Colombia. 2020. p. 9 (6)

- i. **Diseño.** En esta fase se definen las propiedades que se esperan del *software* a desarrollar. Además, se construye un plan de trabajo que puede incluir esquemas de procesos, propuestas de interfaz, criterios de diseño y toda documentación pertinente. Para que esta etapa sea eficaz, es necesario realizar un análisis previo del sistema, establecer con claridad la arquitectura del *software*, diseñar la base de datos incluyendo sus relaciones y dependencias, y estructurar el diseño general del sistema.
- ii. **Implementación.** Esta etapa abarca el desarrollo del *software* como tal. Esto implica la creación del código en función a los objetivos o tareas que hayan sido definidos durante la etapa del diseño. Esto puede implicar de manera adicional la ejecución de pruebas de manera local con el objetivo de validar el correcto funcionamiento de la solución desarrollada.
- iii. **Prueba.** En esta etapa, las características de la solución implementada son examinadas y puestos a prueba de manera minuciosa con el fin de validar la calidad, la presencia de errores en el *software* y el cumplimiento de requisitos solicitados por la parte interesada. En el caso de encontrar observaciones debido a defectos u otros, el equipo de desarrollo subsana las observaciones comunicadas de manera inmediata para que el código sea implementado en producción lo antes posible.
- iv. **Evolución.** En la etapa de evolución los comentarios de los usuarios son recibidos y analizados con el fin de efectivizar el trabajo del equipo y fortalecer los aspectos que faltan mejorar. Posteriormente, se reinicia el ciclo de desarrollo, prueba e incorporación de nuevas

funcionalidades, ya sea en forma de actualizaciones, ajustes o mejoras, lo cual incluye tareas de mantenimiento y asistencia técnica.

Al momento de redactar este documento, el autor ejerce el cargo de analista de control de calidad de sistemas en el departamento de sistemas de la Caja Municipal de Huancayo. El rol del analista de control de calidad se centra en garantizar que los productos entregados cumplan con los criterios de calidad establecidos, respondiendo adecuadamente a las expectativas y requerimientos de los usuarios finales. Esta función está directamente vinculada con la fase de pruebas dentro del ciclo de vida del *software* descrito anteriormente.

“Las pruebas de *software* son un elemento imprescindible y crítico para la validación de un producto de *software*. Es por esto por lo que necesitan apoyarse en estándares que revisan los aspectos fundamentales que debe considerar todo proceso de pruebas” (7).

Con el fin de garantizar la calidad del *software* durante la fase de pruebas, el analista de calidad cumple diversas funciones, entre las cuales se pueden mencionar las siguientes:

- i. **Planificación de pruebas.** Desarrollar estrategias y casos de prueba detallados después de un análisis efectivo en el alcance del requerimiento solicitado por los usuarios con el fin de evaluar la funcionalidad, rendimiento, usabilidad, seguridad y compatibilidad del *software*.
- ii. **Ejecución de pruebas.** Realizar pruebas manuales de tipo estático y funcional para identificar y documentar defectos, errores, malas prácticas de desarrollo, falta de cumplimiento en los estándares utilizando diferentes técnicas y herramientas de prueba.
- iii. **Análisis de resultados.** Analizar las respuestas obtenidas de las pruebas estipuladas para identificar patrones, tendencias y áreas de mejora en el *software*, y, de esta manera, proporcionar una retroalimentación precisa y adecuada a los analistas de desarrollo, analistas funcionales o equipo de solución (equipo que atiende un requerimiento).
- iv. **Documentación.** Generar informes detallados de pruebas que describan y evidencien los hallazgos, resultados, problemas y soluciones propuestas durante el proceso de control de calidad.
- v. **Colaboración interdepartamental.** Trabajar en estrecha colaboración con los analistas de desarrollo, analistas funcionales, usuarios expertos pertenecientes a otros departamentos, y miembros del equipo de despliegues y producción con el fin de garantizar la calidad del producto en todas las etapas del ciclo de vida del *software*.
- vi. **Seguimiento y monitoreo posdespliegue.** Realizar periódicamente un control de los productos que se encuentren en producción para asegurar el correcto funcionamiento de estos manteniendo una coordinación con los usuarios finales hasta obtener una confirmación de su parte sobre el desempeño esperado del producto.

- vii. **Mejora continua.** Contribuir al proceso de mejora continua del Control de Calidad de Software, proponiendo y desarrollando nuevas técnicas, herramientas y prácticas para optimizar la eficiencia y efectividad de las pruebas.

En resumen, el papel que desempeña el analista de control de calidad dentro de la Caja Huancayo es crucial para garantizar la calidad dentro del ciclo de vida del *software*. Asimismo, las responsabilidades de este cargo abarcan diferentes habilidades las cuales contribuyen todas a entregar un producto de excelencia para cubrir las necesidades de las áreas usuarias una vez puesto en funcionamiento. Al tratarse de una empresa financiera como la Caja Huancayo, la cual presenta un crecimiento rápido, cualquier error puede verse reflejado en pérdidas para la empresa además de posibles sanciones impuestas por entes reguladores, riesgo reputacional por servicio brindado de manera incorrecta, o disminución del crecimiento capital. Sin embargo, si los analistas de control de calidad de la Caja Huancayo logran cimentar sus labores en diferentes técnicas, herramientas, normas internacionales y buenas prácticas, aseguran que los productos de *software* sean fiables, seguros y satisfactorios para los usuarios finales y, como consecuencia, para la organización en sí.

2. CAPÍTULO II

INFORMACIÓN GENERAL SOBRE LAS ACTIVIDADES PROFESIONALES

2.1. Diagnóstico Situacional

El análisis situacional de una entidad como la Caja Huancayo constituye un procedimiento que facilita la identificación, descripción, estudio y valoración del estado actual de la organización (Sy, 2020, s. p.). Bajo esta definición, se detalla a continuación un análisis detallado del estado actual de la empresa, evaluando su infraestructura tecnológica, procesos, recursos y capacidades permitiendo, de esta manera, identificar las fortalezas, debilidades, oportunidades y amenazas. Este diagnóstico tiene como propósito reconocer oportunidades de mejora o fortalezas dentro de la empresa.

A continuación, se detalla el diagnóstico situacional del Departamento de Sistemas de la Caja Huancayo reflejado a través del método FODA:



Figura 3. Análisis FODA del Dpto. de Sistemas de la Caja Huancayo

Fuente: Elaboración propia

2.2. Detección de Oportunidades o Necesidades en el Campo Profesional

En un entorno empresarial cada vez más competitivo, la innovación y la rápida adaptación al cambio son fundamentales para el éxito. Bajo lo expuesto y analizado en los capítulos anteriores del presente documento es posible identificar diferentes oportunidades de mejora con el fin de potenciar el rendimiento del departamento de sistemas de la Caja Huancayo y asegurar la satisfacción de los clientes internos. Dentro de las oportunidades identificadas, se ha optado por tratar la automatización de pruebas en el área de control de calidad de sistemas puesto que es el área donde el autor del presente trabajo desempeña sus actividades laborales.

2.2.1. Metodologías ágiles

En el ámbito del desarrollo de *software*, la calidad se ha convertido en un factor crucial para garantizar el éxito y la satisfacción del cliente. Las pruebas de *software* desempeñan un papel fundamental en este proceso, al asegurar que las aplicaciones sean funcionales, seguras y eficientes. Sin embargo, para lograr una gestión efectiva de las pruebas y asegurar la consistencia en los resultados, es necesario adoptar metodologías y marcos de trabajo ágiles y estandarizados que faciliten la organización y ejecución de las pruebas de manera sistemática.

El proceso de atención a los requerimientos de las áreas usuarias en la Caja Huancayo puede verse ralentizado por el seguimiento de procesos normativos estipulados por la misma empresa con el fin de regular los procedimientos que se llevan a cabo. Aun cuando la aplicación de estos procesos está justificada, la falta de actualización en la aplicación de las mismas genera un estancamiento en el desempeño de funciones. La agilidad es la capacidad de responder al cambio y adaptarse a este de manera rápida con el fin de conseguir un resultado beneficioso para la organización.

Scrum

Scrum es un marco de trabajo ágil que ayuda a las organizaciones “a generar valor a través de soluciones adaptables para problemas complejos” (8).

El marco de trabajo que propone la metodología de Scrum “es incompleto de manera intencional” (8), de esta manera solo se establecen las partes necesarias en lugar de indicaciones a detalle con el fin de que las personas que lo utilizan definan de acuerdo con su realidad cómo guiarán sus interacciones.

El empirismo en el cual se basa el Scrum afirma que “el conocimiento proviene de la experiencia y de la toma de decisiones con base en lo observado” (8), con el fin de enfocarse en lo esencial.

Pilares empíricos de Scrum

- a) **Transparencia.** El proceso de trabajo debe ser evidente y visible tanto para el que realiza el trabajo como para el que recibe el producto de este trabajo. Las decisiones importantes se basan en el estado reflejado en estas tareas.
- b) **Inspección.** Las herramientas y el progreso hacia los objetivos deben ser inspeccionado de manera periódica y diligente con el fin de identificar impedimentos.
- c) **Adaptación.** El proceso aplicado para la implementación de un producto puede ajustarse si el resultado del producto no es aceptable o si se encuentra lejos de los límites aceptables. El ajuste se aplica lo antes posible con el fin de minimizar costos y tiempos.

Valores de Scrum

El equipo de trabajo se compromete a incorporar los siguientes valores en su labor. Al hacerlo, los pilares empíricos de Scrum toman fuerza y generan confianza.

- Compromiso
- Foco
- Franqueza
- Respeto
- Coraje.

Roles de Scrum

“La unidad fundamental de Scrum es un pequeño equipo de personas, un Scrum Team” (8). Dentro de este equipo no hay jerarquías, sino que funciona como una unidad de profesionales que tienen el mismo objetivo.

Un Scrum Team consta de lo siguiente:

- *Developers.* Personas que se comprometen a crear los aspectos de un incremento válido en cada periodo de Scrum.
- *Product owner.* Es el responsable de generar el valor del producto que resulta del trabajo del equipo.
- *Scrum master.* Es el responsable de establecer el marco de trabajo en el equipo de manera correcta.

2.2.2. Automatización de pruebas

A pesar de la organización y gran estructura del departamento de sistemas de la Caja Huancayo, la adopción de nuevas tecnologías a los flujos de operaciones que se vienen trabajando es un proceso

lento que requiere de mucha evaluación antes de ser implementado. Esto genera que algunas actividades no se lleven a la velocidad que podrían llevarse. Un ejemplo de esto son las pruebas manuales las cuales se ejecutan en el área de control de calidad para todo tipo de pruebas sin importar el entorno de la aplicación.

La automatización de pruebas se presenta como una efectiva solución para la optimización del proceso actualmente manejado de control de calidad del departamento de sistemas de la Caja Huancayo. El hecho de implementar el presente marco de pruebas automatizadas propuesto permitiría lo siguiente:

- a) Reducción de errores. Los errores humanos se minimizan al ejecutar pruebas de manera automatizada y, a la vez, se garantiza una mayor cobertura sobre los casos de prueba abarcados.
- b) Incremento de eficiencia. Las pruebas pueden ejecutarse de manera rápida y en diferentes entornos cuando son automatizadas. Esto reduce significativamente el tiempo total dedicado las pruebas dentro del ciclo de desarrollo.
- c) Facilitación de pruebas de regresión. Cuando se realicen correcciones y cambios en el código, las pruebas automatizadas serán ejecutadas, asegurando de esta manera que las nuevas modificaciones no introduzcan fallos en funciones existentes.
- d) Enfoque en pruebas complejas. Al reducir los tiempos de las pruebas manuales, los analistas de control de calidad pueden enfocarse en pruebas más exploratorias y a una profundidad que requieren juicio humano y que no podrías ser identificadas con la automatización de pruebas.

La automatización de pruebas es, hoy en día, un fundamento esencial dentro del desarrollo de *software*. Este ofrece, como ya se ha descrito, la posibilidad de mejorar significativamente la calidad de los procesos de prueba y la eficiencia de los analistas de control de calidad. A medida que la Caja Huancayo busque satisfacer las crecientes demandas de los clientes, se vuelve indispensable el uso de la automatización de pruebas. Dentro del presente trabajo, se explorarán los resultados esperados de la adopción de las pruebas automatizadas en nuestro departamento desde la disminución de errores, el incremento en la velocidad de entregas de los productos, hasta la eficiencia en la coordinación y colaboración del equipo. Estos beneficios no solo afectan positivamente la calidad del producto final, sino que también mejoran la forma en que un equipo de desarrollo opera. A continuación, se detallan las expectativas que guiarán la implementación y la evaluación del presente proceso:

- Incrementar la calidad del *software* y la satisfacción de clientes.
- Aumentar la velocidad de entrega de productos al disminuir el tiempo de revisión de los casos de prueba.
- Reducir costos a largo plazo al disminuir el tiempo dedicado a las pruebas manuales.

2.2.3. Responsables de monitoreo posproducción

Dentro de la presente organización, el equipo de desarrollo y pruebas son quienes se encargan de llevar la implementación de los requerimientos solicitados por las áreas usuarias, ejecutar las pruebas para corroborar el correcto funcionamiento de las soluciones, enviar los entregables correspondientes al equipo de producción para su despliegue y realizar el monitoreo del *software* una vez puesto en producción.

Ahora, dependiendo de la complejidad de la aplicación puesta en producción, el seguimiento puede tomar horas para soluciones simples y de ejecución inmediata, o meses si se desean validar generación de intereses en cuentas o capitalizaciones, teniendo en cuenta el rubro de la empresa descrita. Esta actividad, como tarea adicional de las labores del equipo de desarrollo y pruebas, puede restar eficiencia al equipo en la implementación de nuevo *software* y ejecución de pruebas a nuevos requerimientos.

Con el objetivo de optimizar el trabajo del equipo de desarrollo y pruebas, y, a la vez, mejorar la calidad de los monitoreos posterior al despliegue, se pueden asignar roles dedicados al seguimiento de soluciones posterior a la puesta en producción. El brindar esta asignación a un rol específico permitiría un monitoreo más efectivo y especializado, así también, liberaría a desarrollo para centrarse en la entrega de valor continuo para los nuevos productos requeridos por las áreas usuarias.

Por otra parte, de acuerdo con la norma ISO/IEC/IEEE 29119-2:2013, en caso de existir ajustes, correcciones o mejoras al *software* posterior al despliegue, es el equipo de desarrollo y prueba quien debe estar involucrado para brindar una solución efectiva y rápida a las observaciones que se presenten.

Beneficios

- a) Foco del equipo de desarrollo y prueba. Permite al equipo concentrar sus esfuerzos en su principal actividad sobre innovar, desarrollar y validar el funcionamiento, lo cual mejora la productividad y minimiza las fallas.
- b) Especialización. Un rol dedicado de manera exclusiva al monitoreo puede requerir de habilidades específicas en performance, seguridad, observación, entre otros, lo cual eleva la calidad del seguimiento realizado.
- c) Tiempo de respuesta. La detección de fallas y errores se realiza en un menor tiempo cuando se tiene un personal exclusivo para esta actividad. Así también el tiempo de respuesta ante los incidentes disminuye considerablemente.
- d) Mejora continua. Bajo un marco brindado, el personal de monitoreo puede generar retroalimentación útil para elevar la calidad de trabajo de parte del personal de desarrollo como de control de calidad.

2.3. Planteamiento del Problema

El análisis situacional del Departamento de Sistemas de la Caja Huancayo ha permitido identificar diversas fortalezas y debilidades en cuanto a infraestructura tecnológica, procesos y recursos disponibles. Sin embargo, se evidencia una falta de estandarización y actualización en la ejecución del proceso de Control de Calidad de Software, lo cual genera ralentizaciones y dificulta la adaptación ágil a los cambios necesarios para mantener la competitividad y satisfacción de los clientes internos.

En particular, el proceso actual de pruebas de *software* se realiza principalmente de forma manual, sin una automatización adecuada que permita optimizar los tiempos y minimizar errores humanos. Además, la sobrecarga del equipo de desarrollo y pruebas en actividades de monitoreo posproducción afecta la eficiencia en la entrega continua de productos de calidad.

Esta situación plantea la necesidad de implementar metodologías ágiles y adoptar la automatización de pruebas, así como definir roles especializados para el monitoreo posterior al despliegue, con el fin de mejorar la eficiencia y efectividad del proceso de control de calidad en la organización.

A. Problema general

La ausencia de procesos estandarizados y automatizados en el control de calidad de sistemas en la Caja Huancayo limita la eficiencia, la calidad del *software* entregado y la capacidad de respuesta ante incidentes posproducción.

B. Problemas específicos

- ¿En qué medida el proceso actual de pruebas se encuentra alineado con las normas internacionales ISO/IEC/IEEE 29119 e ISTQB?
- ¿Cómo influye la falta de automatización en el desempeño y tiempos del área de control de calidad?
- ¿Qué impacto tiene la ausencia de un rol dedicado al monitoreo posproducción en la eficiencia del equipo de desarrollo y pruebas?
- ¿Cómo puede la implementación de metodologías ágiles, como Scrum, contribuir a la mejora continua del proceso de control de calidad?

2.4. Objetivos de la Actividad Profesional

A. Objetivo general

Describir las diferentes labores realizadas en el Área de Control de Calidad de Sistemas de la Caja Huancayo desde marzo de 2023 hasta la fecha, y cómo el conocimiento adquirido en la carrera de Ingeniería de Sistemas e Informática se ha aplicado en dichas actividades.

B. Objetivos específicos

- Comprobar que el proceso de pruebas llevado a cabo en el Área de Control de Calidad de Sistemas de la Caja Huancayo, desde marzo de 2023 hasta la fecha, está alineado con las normas internacionales ISO/IEC/IEEE 29119 y el estándar ISTQB, aplicando las mejores prácticas en cada fase del proceso.
- Documentar la aplicación del sistema mediante la creación de historias de usuario en el entorno laboral de la Caja Huancayo, desde marzo de 2023 hasta la fecha, explorando funcionalidades, elaborando planes de prueba, escenarios y casos de prueba correspondientes.
- Identificar los factores internos que han afectado o retrasado la ejecución de las labores en el Área de Control de Calidad de Sistemas de la Caja Huancayo, durante el periodo comprendido desde marzo de 2023 hasta la fecha.

2.5. Justificación de la Actividad Profesional

El proceso de control de calidad para el *software* es el que certifica de manera definitiva que el producto cumple con los requisitos de un modo satisfactorio para que este pueda ser puesto en producción. Es por esta razón que las tareas dentro del área de control de calidad deben estar bien organizadas y correctamente fundadas durante todo el ciclo de vida del *software*. La aplicación de buenas prácticas, junto con el uso de herramientas adecuadas, proporciona lo requerido para las tareas correspondientes al Control de Calidad de Software se lleven a cabo de manera efectiva elevando la calidad de los productos implementados por el departamento de sistemas.

La justificación para realizar la documentación del presente trabajo de suficiencia profesional es debido a que es posible identificar y describir procesos dentro de la empresa que están alineados a las normas internacionales de pruebas y calidad las cuales brindan resultados beneficiosos para la organización. Así también, se han identificado oportunidades de mejora los cuales afectan o ralentizan las actividades que desempeñan los analistas de control de calidad de sistemas dentro de la Caja Huancayo.

Se espera, a través de los resultados del presente trabajo, identificar de manera sólida las fortalezas que se tiene en el área de control de calidad de sistemas, así como los aspectos a optimizar desde el punto de vista del autor.

2.6. Tipo y Enfoque de Investigación

El presente trabajo tiene un tipo de investigación aplicada y descriptiva, dado que se enfoca en la aplicación práctica del conocimiento adquirido en la carrera de Ingeniería de Sistemas e Informática para analizar y mejorar el proceso de control de calidad en la Caja Huancayo. Se describen los procesos

actuales, se identifican oportunidades de mejora y se aplican soluciones específicas para optimizar las labores del área de aseguramiento de calidad.

El enfoque es cualitativo y cuantitativo. Por un lado, se realiza un análisis cualitativo del contexto, procesos y metodologías aplicadas, y, por otro lado, se incorporan datos cuantitativos sobre la ejecución de pruebas, incidencias detectadas y resultados obtenidos para evidenciar mejoras y logros concretos.

2.7. Resultados Esperados

A través del presente trabajo se espera evidenciar la aplicación práctica del conocimiento adquirido durante la carrera de Ingeniería de Sistemas e Informática, en el desarrollo de actividades propias del rol de analista de Control de Calidad de Sistemas en la Caja Huancayo.

Como resultado de este ejercicio profesional, se busca comprobar que el proceso de pruebas actualmente implementado en el Área de Control de Calidad de Sistemas se encuentra alineado con estándares internacionales, específicamente con la norma ISO/IEC/IEEE 29119 y los principios establecidos por el ISTQB. Esta alineación permitirá mejorar la trazabilidad de los casos de prueba y los defectos identificados, facilitando una gestión más eficaz de los resultados y asegurando la cobertura de los requisitos funcionales y no funcionales del sistema. Asimismo, se anticipa una mejora en la resolución de defectos y en la retroalimentación ofrecida al equipo de desarrollo.

También se espera documentar la aplicación del conocimiento técnico en la elaboración de historias de usuario, la planificación de pruebas y la creación de casos de prueba, así como en el uso de herramientas especializadas. La adopción de metodologías estandarizadas permitirá incrementar la cobertura de las pruebas, incluyendo pruebas funcionales, de rendimiento, de seguridad, entre otras, a fin de garantizar que se contemplen todos los escenarios críticos del *software*. Adicionalmente, se espera que la estandarización contribuya a optimizar el uso del tiempo y los recursos del equipo de pruebas.

Finalmente, se busca identificar factores internos que estén afectando o retrasando la ejecución eficiente de las labores en el área de calidad. Detectar estas limitaciones permitirá proponer acciones de mejora orientadas a optimizar los procesos, aumentar la eficiencia del equipo y elevar la calidad del *software* entregado.

3. CAPÍTULO III MARCO TEÓRICO

3.1. Fundamentos Teóricos de las Actividades Desarrolladas

3.1.1. Proceso de verificación de *software* en la entidad financiera Caja Huancayo

Para una correcta validación de un producto de *software*, las pruebas son una etapa crítica y fundamental. Para garantizar la calidad de sus productos, la Caja Huancayo basa su procedimiento de pruebas en estándares internacionales que establecen criterios esenciales que deben ser tomados en cuenta durante dicho proceso.

ISO/IEC 29119 (ESTÁNDAR INTERNACIONAL PARA PRUEBAS DE SOFTWARE)

El estándar internacional ISO/IEC 29119 brinda “un marco de referencia común para las pruebas de *software* cubriendo todos los aspectos del ciclo de vida” (7).

Con el alcance de la presente norma, se propone crear un estándar de pruebas de *software* que sea aplicable para todos los tipos de productos de *software*. Asimismo, el ISO/IEC29119 tiene como propósito “unificar e integrar el corpus actualmente fragmentado de literatura normativa sobre pruebas que actualmente ofrecen tres organismos de normalización distintos: VSI, IEEE e ISO/IEC JTC 1/SC 7. El resultado del proyecto será un tratamiento coherente y unificado por las tres organizaciones” (9).

a) Parte 1. Definiciones y vocabulario

Con esta primera parte, se ofrece una perspectiva general sobre la norma y los conceptos fundamentales relacionados con las pruebas de *software*. Además, presenta un glosario de términos y definiciones que facilitan la comprensión y aplicación de las pruebas a lo largo de las distintas fases del ciclo de vida del *software*.

Entre los conceptos sobre las pruebas de *software* se presentan los siguientes:

- Las pruebas de *software* en las organizaciones
- Los procesos de prueba en los ciclos de vida (visión general)
- Pruebas basadas en riesgos
- Estrategias de prueba
- Automatización de las pruebas
- Problemas en la gestión.

b) Parte 2. Proceso de prueba

La parte dos permite definir las pruebas de proceso genérico con un modelo a usar dentro de cualquier desarrollo de *software*.

Este modelo se basa en un proceso de prueba que abarca tres capas:

- Procesos de prueba de la organización
- Procesos de gestión de las pruebas
- Procesos de pruebas dinámicas.



Figura 4. Proceso de Pruebas de Software en 3 capas

Fuente: ORGANISMO de Normalización en España (1º 2015: San Sebastián). ISO/IEC/IEEE 29119 El nuevo estándar internacional para pruebas de *software*. AENOR. 2015. p. 15 (9)

c) Parte 3. Documentación de la prueba

Dentro de esta parte las plantillas que pueden ser utilizadas para la documentación son definidas. Estas plantillas serán usadas en todo el ciclo de vida del *software* y son un producto de los procesos de la parte dos.

1. Documentación de proceso de pruebas de la organización:
 - Política de pruebas
 - Estrategia de la organización para las pruebas
2. Documentación del proceso de gestión de pruebas:
 - Plan de pruebas
 - Reporte de estado de pruebas

3. Documentación del proceso de pruebas dinámicas:

- Especificación de caso de prueba
- Especificación del procedimiento de prueba
- Requerimientos de datos de prueba
- Requerimientos en los entornos y ambientes de prueba
- Resultados obtenidos en las pruebas
- Registro de las ejecuciones de prueba
- Informe de incidentes identificados en las pruebas

d) Parte 4. Técnicas de prueba

La cuarta sección presenta diversas técnicas para diseñar pruebas de *software*, también conocidas como métodos o estrategias para la creación de casos de prueba. Estas técnicas pueden aplicarse tanto en las fases de diseño como durante la implementación de pruebas, en cualquier organización que se dedique al desarrollo de *software*. Los métodos para el diseño de pruebas se fundamentan en las especificaciones, la estructura o la experiencia previa (9).

1. Técnicas basadas en las especificaciones:

- Evaluación de valores límite
- Prueba mediante tablas de decisión
- Análisis de escenarios de prueba
- Pruebas de transición entre estados
- Métodos adicionales

2. Técnicas fundamentadas en la estructura:

- Cobertura de instrucciones en el código
- Pruebas de bifurcaciones
- Métodos adicionales

3. Técnicas fundamentadas en la experiencia:

- Hipótesis de fallos

e) Parte 5. Pruebas basadas en palabras clave

Asimismo, de acuerdo con la parte 2 de la norma de la presente ISO 29119 (ISO/IEC/IEEE 29119-2:2013) se describe cómo debe gestionarse de manera completa el ciclo de pruebas mencionando también que el equipo de desarrollo y prueba puede involucrarse en la verificación de correcciones o cambios posterior a la puesta en producción del producto implementado.

ISTQB (International Software Testing Qualifications Board)

Esta organización internacional certifica a profesionales en pruebas de *software*. Establece un “esquema internacionalmente conocido creado por una red global de expertos” (10) en lo concerniente a pruebas dentro del ciclo de vida del *software*. Las definiciones de la presente certificación basan sus definiciones en siete principios del *testing de software*:

- i. Las pruebas muestran la presencia de defectos, no su ausencia.

La ejecución de pruebas reduce la probabilidad de errores no detectados en el *software*, pero es imposible demostrar la ausencia de estos.

- ii. Las pruebas exhaustivas son imposibles.

La posibilidad de probar absolutamente todo es un objetivo inalcanzable. Es posible concentrar todos los esfuerzos en las pruebas para optimizar el alcance de estas a través de diferentes técnicas, análisis y priorización.

Como resulta imposible validar todas las combinaciones de entradas y salidas en un sistema, se requiere identificar las funciones principales de la solución y asegurar su validación. Sin embargo, se debe asumir que dentro del margen de error en las pruebas a realizar se ubican algunos riesgos.

- iii. Las pruebas tempranas ahorran tiempo y dinero.

Las pruebas de *software* tienen que iniciar en las primeras fases del ciclo de vida del *software*. Si un error es detectado en las fases iniciales, el costo de la solución es más barato y rápido que cuando un fallo debe ser solucionado posterior a su puesta en producción.

- iv. La agrupación de defectos.

Se deben priorizar las partes fundamentales de la aplicación con el fin de minimizar el número de errores y el posible riesgo puesto que la mayoría de los defectos se concentran en un número reducido de módulos del *software*.

- v. La paradoja del pesticida.

Si las mismas pruebas son empleadas para encontrar defectos en el sistema, la probabilidad de encontrar nuevas fallas es menor. Los casos de prueba se deben modificar, junto con los datos de prueba, para la detección de nuevos errores.

- vi. Las pruebas dependen del contexto.

Las pruebas que se realizan pueden variar de acuerdo con el contexto de uso puesto que distintos proyectos utilizan distintas aplicaciones es requerido realizar diferentes pruebas.

vii. La ilusión de ausencia de errores es peligrosa.

Al realizar las pruebas de *software*, es reconocido que no existen las pruebas exhaustivas (principio número dos). “No obstante, la detección e identificación de los posibles errores en el plan de pruebas demuestra que el plan no cuenta con errores (11).

3.1.2. Proceso de ejecución de pruebas de *software*

Para definir correctamente el proceso de pruebas de *software*, es necesario entender que este no solo consiste en la ejecución de pruebas, sino que “existen otras actividades de pruebas, antes, durante y después de la ejecución” (7).

- Organizar y supervisar el proceso.
- Seleccionar las condiciones bajo las cuales se realizarán las pruebas.
- Crear los casos para la ejecución de pruebas.
- Verificar los resultados obtenidos durante las pruebas.
- Analizar los criterios para determinar la finalización de las pruebas.
- Documentar y reportar los resultados de las pruebas.
- Coordinar con las áreas usuarias la realización de pruebas de aceptación.
- Ejecutar las actividades finales para concluir la fase de pruebas.

3.1.3. Proceso fundamental del *testing*

a) Planificación y control

- i. Comprender los objetivos del usuario, el proyecto y los riesgos en las tareas de *testing*.
- ii. Fijar el alcance de las pruebas, así como identificar los objetivos que tendrá el *testing*.
- iii. Aplicar estrategias de prueba.
- iv. Programar las tareas del diseño de pruebas.

b) Análisis y diseño

- i. Examinar los componentes fundamentales necesarios para llevar a cabo las pruebas, incluyendo diseños, especificaciones, arquitecturas e interfaces.
- ii. Identificar las condiciones para las pruebas junto con los datos requeridos en base al análisis del punto anterior.
- iii. Determinar las herramientas que sean necesarias para la ejecución de las pruebas y el entorno requerido.

c) *Ejecución*

- i. Ejecutar las pruebas manualmente.
- ii. Documentar los resultados obtenidos durante la ejecución.
- iii. Comunicar los errores detectados durante las pruebas.
- iv. Contrastar los resultados reales con los previstos.
- v. Reportar cualquier incidente o diferencia encontrada.
- vi. Repetir las pruebas cuando se realicen correcciones en el desarrollo, si se han identificado observaciones.
- vii. Evaluar los criterios de salida:
 - Evaluar si se requerirán más pruebas.
 - Registrar un resumen de las pruebas ejecutadas.

d) *Cierre de pruebas*

- i. Gestión de aprobaciones requeridas como control para los despliegues de *software* a producción.
- ii. Envío de entregables finales de la solución implementada al equipo de producción para su despliegue.
- iii. Recolectar la información sobre las pruebas completadas para su consolidación y reporte.
- iv. Verificar que las correcciones hayan sido cargadas de manera correcta en el sistema como nuevos entregables.
- v. Publicar el material generado en las pruebas como evidencia del trabajo realizado.
- vi. Evaluar el resultado de las tareas de pruebas y analizar las lecciones aprendidas durante el *testing* realizado.

3.1.4. Validación de programas informáticos

Las evaluaciones del *software* consisten en “un proceso de la ingeniería por el cual se controla que un sistema cumpla con sus requisitos funcionales y funcione correctamente adecuándolo a unos estándares de calidad y fiabilidad” (7).

Este proceso permite revisar el sistema a validar bajo ciertas condiciones definidas de manera clara y explícita “para garantizar la correcta funcionalidad e interacción con el usuario final y asegurar la calidad del *software*” (6).

El fundamento de las pruebas está basado en siete principios (descritos anteriormente en la información sobre ISTQB del presente documento). “Estos principios buscan recoger ciertas creencias que se tiene con respecto a las pruebas y aclararlas de acuerdo con la realidad de la profesión” (12).

A continuación, se nombras estos siete principios mencionados con más detalle en la información sobre ISTQB del presente documento:

- a) Las pruebas muestran la presencia de defectos, no su ausencia.
- b) Las pruebas exhaustivas son imposibles.
- c) Las pruebas tempranas ahorran tiempo y dinero.
- d) La agrupación de defectos.
- e) La paradoja del pesticida.
- f) Las pruebas dependen del contexto.
- g) La ilusión de ausencia de errores es peligrosa.

3.1.5. Propósitos de la evaluación del *software*

Los objetivos fundamentales al realizar pruebas de *software* incluyen los siguientes:

- Reducir los riesgos.

Disminuir las posibles complicaciones durante la operación del *software* y “garantizar que el *software* [...] tenga el comportamiento adecuado y no genere problemas que puedan afectar la interacción del usuario final con el mismo” (6).

- Garantizar la calidad del *software*.

Asegurar la satisfacción del cliente con el producto final teniendo en cuenta el funcionamiento de manera correcta, una interacción adecuada y verificando que las respuestas sean las esperadas por el usuario; en resumen, que cumple con el objetivo para el que fue creado.

- Identificar defectos

Dado el error humano se pueden presentar desperfectos en el *software*, los cuales deben ser encontrados y corregidos antes de ser llevado al usuario final. Para esto “es necesario llevar a cabo un proceso que permita interactuar con el *software* antes de ser expuesto al cliente para realizar las verificaciones pertinentes en cuanto a su correcto comportamiento y cumplimiento de criterios” (6).

- Identificar cambios no intencionados

Al efectuar modificaciones o mejoras en la implementación, pueden surgir cambios que afecten el funcionamiento del *software*. Por esta razón, es esencial llevar a cabo pruebas no solo sobre los ajustes recientes, sino también para asegurar que el sistema completo continúe operando correctamente y cumpliendo con sus funciones y objetivos establecidos.

3.1.6. Enfoque de las pruebas de *software*

Existen diferentes motivos para que un proceso de ejecución de pruebas pueda ser llevado a cabo. Esto puede depender de los objetivos, ajustes o mantenimiento del *software*, como también de sus funcionalidades. Los enfoques para las pruebas son los siguientes:

- a) Enfoque analítico. Centrado en la identificación y gestión de riesgos.
- b) Enfoque basado en modelos. Emplea datos estadísticos como la frecuencia de errores, patrones de uso, entre otros.
- c) Enfoque metódico. Basado en experiencia, fallos, listas de comprobación y características.
- d) Enfoque de proceso. Basado en metodologías ágiles.
- e) Enfoque dinámico. Basado en pruebas exploratorias.
- f) Enfoque consultivo. Basado en recomendaciones y orientaciones de expertos en las áreas a probar.
- g) Enfoque antiregresión: Basado en reutilizar el material de pruebas.

3.1.7. Técnicas aplicadas al diseño de casos de pruebas

Al tener claros los objetivos del *software* a implementar, junto con la evaluación de los riesgos del sistema, es posible empezar con la planificación de pruebas. Para esto, “se seleccionan las técnicas de diseño, tipos de pruebas a aplicar y se definen los criterios de entrada y salida” (6). Para la selección de la técnica a emplear es necesario evaluar lo siguiente:

- Tipo del sistema
- Tipo de riesgo
- Nivel de riesgo
- Estándares normativos
- Requisitos del cliente
- Objetivo de la prueba
- Documentación disponible

- Ciclo de vida de desarrollo
- Modelos de caso de uso
- Experiencia en los defectos previamente detectados
- Análisis del impacto (pruebas de regresión necesarias, cómo se ve afectado el sistema actual frente a los cambios realizados)

Al evaluar los mencionado, se tiene los siguientes métodos para el diseño de casos de prueba según corresponda:

3.1.7.1. Pruebas de caja blanca

Este tipo de pruebas basa su análisis en la estructura interna del sistema. “Los casos de prueba se seleccionan a partir del mismo código, información de diseño de *software*, información técnica de la implementación, arquitectura, entre otros” (6).

3.1.7.2. Pruebas de caja negra

Los casos de prueba para este tipo de técnica se seleccionan a partir de los objetivos o criterios de aceptación determinados previamente por el usuario. Se pueden incluir los siguientes casos:

a) Partición de equivalencia.

Es necesario clasificar los valores de entrada y salida que presentan comportamientos similares, tanto para datos válidos como inválidos. Esta agrupación facilita la detección de inconsistencias en las especificaciones y reduce la cantidad de casos de prueba requeridos. No obstante, “no tiene en cuenta las posibles interacciones entre las condiciones de prueba y no resuelve problemas de estabilidad del *software*” (6).

b) Análisis de límites.

Se deben tener en cuenta los valores máximos y mínimos dentro de particiones válidas e inválidas para detectar posibles fallos en el *software* desarrollado.

c) Tablas de decisión.

Identificar condiciones de entrada y salida del *software* usando los requisitos de este con el cual, al combinar las condiciones, se establecen las reglas.

d) Transición de estado.

Este diagrama facilita la identificación de los diferentes estados que puede presentar el sistema, así como las interacciones válidas e inválidas entre dichos estados. “Un estado representa una situación de un sistema hasta que sea modificado por un evento. Un evento representa una situación exterior que puede ocurrir cuando el sistema se encuentra en determinado estado” (6).

e) Pruebas de casos de uso

Estos representan las interacciones entre el sistema y los usuarios, con el objetivo de generar un producto que aporte valor al usuario. Se toman en cuenta las condiciones previas necesarias para el correcto desarrollo del proceso, así como las condiciones posteriores que garantizan que los resultados obtenidos sean los esperados.

3.1.7.3. Pruebas basadas en la experiencia

Las pruebas de este tipo se desprenden de la intuición o destreza del analista de control de calidad en base a su experiencia con las aplicaciones similares, errores identificados previamente, entre otros. De esta manera se listan de manera anticipada las posibles fallas identificadas que pueden ser atendidas a través de la ejecución de pruebas adecuadas.

3.1.8. Clasificación de pruebas

Se identifican dos categorías principales de pruebas que deben realizarse en todos los casos para garantizar la validación del *software*. Ambos tipos de pruebas son distintos, pero complementarios para lograr un mayor alcance en la ejecución, así como mayor efectividad en el mismo.

3.1.8.1. Pruebas estáticas

Este tipo de pruebas posibilita la detección de errores en el sistema mediante la revisión de la sintaxis del código creado para la solución. Para ello no es necesaria la ejecución del código, sino que es necesaria la inspección del código como tal y un análisis estático.

3.1.8.2. Pruebas dinámicas

Este tipo de pruebas se enfoca en la detección de fallos dentro del *software*. Incluye las siguientes categorías:

- i. Pruebas funcionales. Se verifica que el sistema desempeñe correctamente las funciones definidas según los requerimientos establecidos por el usuario. Las funcionalidades deben satisfacer las necesidades tal como fueron especificadas.
- ii. Pruebas no funcionales. Se debe validar el comportamiento del sistema como los requisitos no funcionales entre los que se encuentran las siguientes características:
 - Facilidad de uso
 - Protección y seguridad
 - Eficiencia y desempeño
 - Estabilidad
 - Robustez

- Disponibilidad
 - Portabilidad
 - Integridad
 - Eficiencia
- iii. Pruebas basadas en la estructura. Se debe evaluar la cobertura de las pruebas para medir su exhaustividad.
- iv. Pruebas de regresión. Se deben validar los cambios sobre las pruebas ya realizadas. Son pruebas repetidas de un sistema ya validado luego de haber realizado cambios o ajustes al mismo.

3.1.9. Niveles en la ejecución de pruebas

El nivel de ejecución en las pruebas de *software* depende del objetivo que se desee alcanzar en las pruebas, las funcionalidades requeridas, el enfoque y las técnicas que se vayan a realizar durante este proceso.

3.1.9.1. Pruebas de componente

También denominadas pruebas unitarias, esta etapa tiene como propósito verificar que cada módulo individual del sistema funcione correctamente. Esto incluye objetos tales como métodos, clases, formularios, entre otros. Los defectos se identifican en los componentes de manera unitaria.

3.1.9.2. Pruebas de integración

En este nivel de pruebas se valida que la interacción entre dos o más componentes del sistema es correcta y cumplen su función en conjunto. Los elementos más habituales para realizar esta validación incluyen la implementación de bases de datos, la infraestructura y las interfaces.

3.1.9.3. Pruebas a nivel del sistema

En el nivel de pruebas de sistema se verifica que el sistema completo opere correctamente en su conjunto, evaluándolo a nivel de producto. Para que esto se lleve a cabo, “el entorno de pruebas debe ser lo más parecido al entorno de producción [...] para minimizar fallas y defectos” (6).

3.1.9.4. Pruebas de aceptación

El objetivo del nivel de pruebas de aceptación es comprobar un que el sistema implementado es funcional y confiable para los usuarios finales. Para este nivel de pruebas se involucra usuarios finales en la participación de pruebas para verificar que el producto cumple con sus necesidades. En estas pruebas se deben reflejar escenarios de uso real que demuestren la funcionalidad y usabilidad del *software* y, de esta manera, se mejora la satisfacción del cliente y aumenta la confianza sobre el producto.

4. CAPÍTULO IV

DESCRIPCIÓN DE LAS ACTIVIDADES PROFESIONALES

Dentro del presente capítulo se detallan y describen las actividades profesionales principales que desempeña un analista de Control de Calidad de Software en la Caja Municipal de Huancayo a la fecha en que este trabajo es redactado a través de la exposición de un caso práctico, el cual permite ilustrar de manera concreta las funciones que se desempeñan, así como las competencias técnicas y metodológicas aplicadas en el entorno laboral.

Para este ejemplo, se presentará la experiencia del proceso de *testing* de una página web de un producto ofrecido por la Caja Huancayo denominado “Cuenta Futuro”. Este producto de ahorro ofrecido por esta organización tiene “la finalidad de promover la cultura de ahorro en menores de edad” (13) (Caja Huancayo, s. p.) incentivando a administrar de manera responsable el uso del dinero para cumplir sus metas. Con el objetivo de impulsar el uso de este producto de una manera más llamativa, el área de negocios solicita la implementación de un portal web dirigido a niños y niñas que busca fomentar el hábito del ahorro mediante una interfaz amigable, intuitiva, segura y pedagógicamente adecuada para su público objetivo.

El análisis se centra particularmente en la etapa de ejecución de pruebas de *software* para la verificación y validación de este nuevo servicio, considerando criterios de calidad, el uso de buenas prácticas, la aplicación del empleo de estándares institucionales, cumplimiento de funcionalidad, usabilidad y accesibilidad. Todos estos son fundamentales para garantizar que la aplicación cumpla con los estándares exigidos tanto por los usuarios finales como por los organismos reguladores. Asimismo, se describirá el enfoque metodológico empleado, las herramientas e instrumentos utilizadas durante la ejecución de pruebas de *software* y los principales hallazgos obtenidos con el fin de evidenciar el impacto del trabajo profesional realizado en el éxito resultante del proyecto.

A continuación, se presenta la arquitectura general del proyecto desarrollado, donde se visualizan los principales componentes involucrados en el sistema: FRONTEND, BACKEND, servicios de API, base de datos y su interacción. El uso de esta arquitectura, sobre todo durante la etapa inicial del ciclo de vida del *software*, es clave para la planificación y ejecución de pruebas de calidad de *software* durante todas las fases del proyecto. Así también una descripción breve de los casos de uso que describen la funcionalidad del presente proyecto.

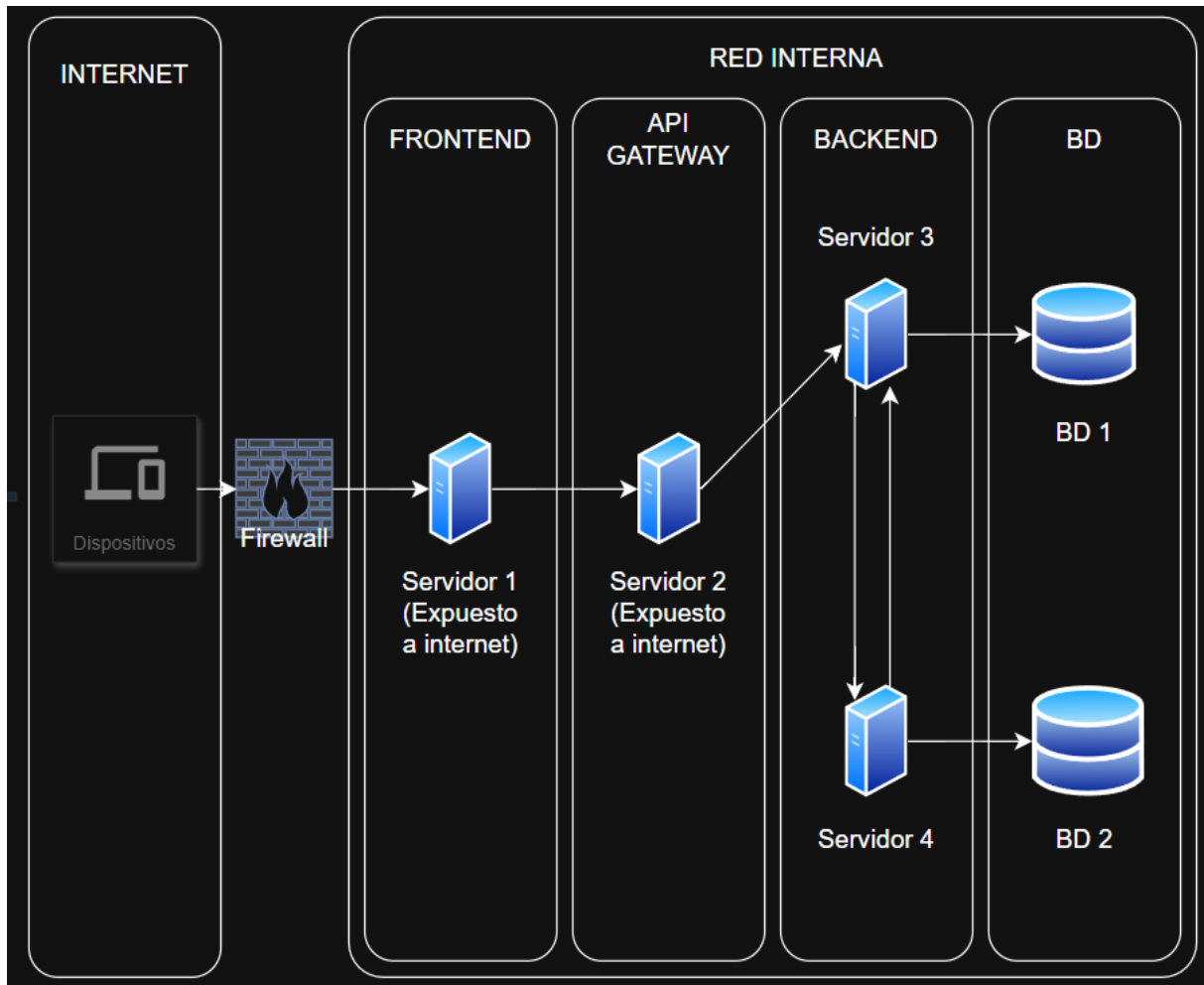


Figura 5. Arquitectura general del proyecto

Fuente: Elaboración propia

N.º	Nombre del caso de uso	Actor principal	Descripción breve
CU01	Registro de clave web nueva	Usuario menor de edad	Permite registrar una clave web para acceder por primera vez a la plataforma.
CU02	Inicio de sesión	Usuario menor de edad	Permite al usuario acceder a la plataforma ingresando su clave web.
CU03	Validación de tarjeta	Sistema	Verifica que la tarjeta ingresada pertenezca a una Cuenta Futuro activa.
CU04	Recuperación de clave web	Usuario menor de edad	Permite recuperar o resetear la clave web mediante verificación.
CU05	Selección de personaje	Usuario menor de edad	Permite elegir un personaje para representar al usuario en la plataforma.
CU06	Selección de habilidad para el personaje	Usuario menor de edad	Selecciona una habilidad especial para el personaje elegido.
CU07	Cargar juegos de la plataforma	Usuario menor de edad	Accede a los juegos disponibles en la plataforma de ahorro.
CU08	Ver datos de la cuenta	Usuario menor de edad	Muestra la información básica de la Cuenta Futuro y del cliente.
CU09	Editar personaje	Usuario menor de edad	Cambia el personaje elegido previamente por otro.
CU10	Editar habilidad	Usuario menor de edad	Cambia la habilidad asignada al personaje.
CU11	Registrar meta de ahorro	Usuario menor de edad	Permite fijar una meta de ahorro (monto, motivo, plazo).
CU12	Ver saldo y movimientos de la cuenta	Usuario menor de edad	Muestra el saldo actual y los movimientos de ingreso y egreso pertenecientes a la Cuenta Futuro.
CU13	Ganar medalla	Sistema	Otorga una medalla al lograr una meta de ahorro en el marco de una campaña.
CU14	Cambiar de cuenta	Usuario menor de edad	Permite cambiar la visualización de la plataforma con otra cuenta activa asociada al cliente sin cerrar la sesión actual.
CU15	Cerrar sesión	Usuario menor de edad	Finaliza la sesión del usuario en la plataforma.

Figura 6. Casos de uso del proyecto

Fuente: Elaboración propia

4.1. Descripción de Actividades Profesionales

4.1.1. Enfoque de las actividades profesionales

El enfoque de las actividades profesionales como analista de Control de Calidad de Software está orientado a garantizar la calidad del *software* desde las primeras etapas del ciclo de vida del proyecto, hasta su puesta en producción. El analista se enfocó en la planificación, diseño, ejecución y seguimiento de pruebas que aseguren el correcto funcionamiento de la página web para cumplir con los requisitos funcionales y no funcionales establecidos.

La labor realizada se alinea a los principios del aseguramiento de la calidad: prevención de defectos, mejora continua, cumplimiento de estándares y satisfacción del usuario final, en este caso de los menores de edad, padres y usuarios pertenecientes al área de negocio de la Caja Huancayo.

4.1.2. Alcance de las actividades profesionales

Las actividades del bachiller como analista de control de calidad abarcaron los siguientes puntos:

- a) Revisión de la especificación del requerimiento
- b) Análisis de la arquitectura propuesta
- c) Revisión de los requerimientos funcionales y técnicos del sistema
- d) Solicitud de ambientes y herramientas requeridas para la ejecución de pruebas
- e) Elaboración del plan de pruebas
- f) Diseño de casos de prueba orientados a la experiencia de usuarios menores de edad
- g) Ejecución manual de pruebas funcionales en módulos clave
- h) Reporte, seguimiento y validación de errores
- i) Coordinación con el equipo de desarrollo
- j) Coordinación con el área usuaria (área de negocio)
- k) Publicación del proyecto simulando un ambiente productivo
- l) Validación final con el área usuaria del producto antes de su despliegue
- m) Gestión y monitoreo de un despliegue controlado
- n) Seguimiento a la etapa de piloto o marcha blanca
- o) Informe sobre resultados de la etapa del piloto
- p) Envío del despliegue a nivel masivo.

4.1.3. Entregables de las actividades profesionales

Durante el desarrollo de la ejecución de pruebas para este proyecto, el analista de control de calidad genera los siguientes entregables:

- a) Plan de pruebas para el sistema web
- b) Casos de prueba documentados por funcionalidad donde se incluyen escenarios ideales y no ideales
- c) Evidencias de pruebas ejecutadas (capturas de pantalla, resultados obtenidos, tiempos de ejecución de procesos)
- d) Reporte de observaciones identificadas registrados en la herramienta de gestión de proyectos Azure DevOps
- e) Informe de envío de la solicitud a pruebas de aceptación por el usuario
- f) Informe de validación final previo a la liberación del producto
- g) Checklist de criterios de aceptación específicos para asegurar la completa validación de estándares institucionales y normas internacionales en el producto evaluado
- h) Plan de despliegue en la etapa del piloto
- i) Plan de despliegue masivo
- j) Plan de contingencia en caso de eventos posproducción
- k) Arquitectura no productiva que se usó con el fin de brindar información para futuros mantenimientos al sistema
- l) Manual de configuración del proyecto dentro de un ambiente no productivo con el fin de brindar información para futuros mantenimientos al sistema
- m) Informe del monitoreo y seguimiento posproducción hasta la estabilización del proyecto

4.2. Aspectos Técnicos de la Actividad Profesional

4.2.1. Metodologías

Para la ejecución de las actividades profesionales, se aplicó la metodología de desarrollo en cascada, un enfoque secuencial y estructurado que permitió organizar el ciclo de vida del proyecto en etapas claramente definidas. Este modelo fue adecuado considerando la naturaleza del proyecto, que requería una planificación detallada, documentación exhaustiva y una validación estructurada de los entregables antes de pasar a la siguiente fase.

Desde el rol de analista de Control de Calidad de Software, el bachiller participó principalmente en las siguientes fases del modelo en cascada:

- i. Análisis de requerimientos. Revisión minuciosa de los requisitos funcionales y no funcionales definidos para la plataforma web, asegurando su claridad, coherencia y viabilidad para ser validados mediante pruebas.
- ii. Diseño del sistema y diseño de pruebas. Desarrollo del plan de pruebas, casos de prueba y escenarios de validación alineados con los documentos de especificación. En esta etapa, se establecieron los criterios de aceptación del sistema.
- iii. Implementación. Aunque esta etapa fue responsabilidad del equipo de desarrollo, el bachiller intervino para preparar el entorno de pruebas y automatizar scripts de prueba en paralelo.
- iv. Verificación (fase de pruebas). Ejecución sistemática de pruebas unitarias, funcionales, de integración y de regresión, con la finalidad de verificar que el sistema cumpla con los requisitos establecidos. Se registraron los errores encontrados y se validaron las correcciones.
- v. Mantenimiento. Una vez desplegado el sistema, el bachiller realizó un monitoreo postproducción para asegurar la estabilidad de la plataforma y documentó hallazgos para futuras mejoras.

Sin embargo, también se tomaron elementos de la metodología ágil Scrum, lo cual implicó la participación activa del analista de control de calidad en:

- Reuniones diarias (*daily meeting*)
- Planificación de Sprints
- Retrospectivas

Este enfoque metodológico permitió una ejecución ordenada y controlada de las pruebas, con entregables claros en cada fase y un seguimiento riguroso de los resultados obtenidos.

4.2.2. Técnicas

Durante el desarrollo del proyecto, se aplicaron diversas técnicas de prueba de *software* con el fin de asegurar la calidad del sistema en sus diferentes niveles. Estas técnicas permitieron evaluar tanto el comportamiento externo como la lógica interna del sistema, adaptándose al tipo de funcionalidades y a los usuarios objetivos de la plataforma.

Las técnicas utilizadas fueron las siguientes:

- a) Pruebas de caja negra. Se evaluó la funcionalidad del sistema sin tener en cuenta la estructura interna del código. Esta técnica permitió verificar que, dadas determinadas entradas, el sistema produjera las salidas esperadas, validando principalmente los formularios de registro,

acceso, juegos educativos, historial de ahorro y los flujos relacionados con el uso del producto financiero “Cuenta Futuro”.

- b) Pruebas de caja blanca. Se revisaron estructuras internas del código, caminos lógicos, condiciones y bucles para asegurar una cobertura completa del código fuente y su correcta ejecución. Esta técnica se aplicó especialmente en funciones críticas relacionadas con la validación de datos, la conexión con el sistema financiero de Caja Huancayo y los cálculos de ahorro.
- c) Pruebas basadas en la experiencia. Se aplicaron pruebas exploratorias y heurísticas basadas en el conocimiento del bachiller sobre plataformas web. Esta técnica permitió detectar errores no previstos en los casos de prueba formales, evaluando escenarios de uso atípicos, posibles errores de interfaz y barreras de usabilidad.
- d) Diseño de pruebas funcionales basadas en casos de uso. Cada módulo fue evaluado con base en sus casos de uso definidos, asegurando que se cumplan los objetivos esperados por el usuario final.
- e) Pruebas de regresión. Se verificó que los cambios o correcciones aplicados en versiones posteriores no afectaran funcionalidades previamente validadas.
- f) Revisión por pares de casos de prueba. Los casos de prueba fueron revisados en conjunto con otros miembros del equipo de calidad para garantizar su coherencia, completitud y alineación con los requerimientos.

Estas técnicas permitieron asegurar la calidad del sistema desde múltiples perspectivas, brindando confianza sobre el funcionamiento correcto y seguro de la plataforma antes de su despliegue final.

4.2.3. Instrumentos

Durante el desarrollo del proyecto y la ejecución de las actividades profesionales, se utilizaron una variedad de instrumentos tecnológicos y documentales que facilitaron la planificación, ejecución y seguimiento del proceso de control de calidad del *software*. Estos instrumentos permitieron validar tanto el funcionamiento técnico del sistema como su adecuación a los requisitos del usuario. A continuación, se describen los principales instrumentos utilizados:

- i. Azure DevOps. Plataforma de gestión de proyectos utilizada para el seguimiento de tareas, administración de planes de prueba, control de versiones y registro de errores (*bugs*). A través de esta herramienta se gestionó el ciclo de vida de las pruebas, desde la planificación hasta la validación de correcciones.

- ii. Postman. Herramienta empleada para ejecutar pruebas sobre las API del sistema. Permitió validar la integridad de los servicios, la autenticación, la lógica de negocio expuesta en el BACKEND, así como el manejo correcto de errores y respuestas ante diferentes entradas.
- iii. Visual Studio Code. Editor de código fuente utilizado para ejecutar el FRONTEND de la aplicación en un entorno local y, posteriormente, generar una publicación no productiva, permitiendo validar el comportamiento visual, los flujos de navegación y la interacción con la interfaz gráfica, particularmente en funcionalidades dirigidas a menores de edad.
- iv. Visual Studio. Entorno de desarrollo usado para compilar, depurar y ejecutar el BACKEND del sistema, desarrollado en tecnologías .NET. Fue esencial para realizar pruebas técnicas y funcionales en un entorno controlado, antes de la integración con el FRONTEND.
- v. SQL Server Management Studio (SSMS). Herramienta utilizada para acceder a la base de datos del sistema, realizar consultas SQL, validar la correcta inserción, actualización y recuperación de datos, así como asegurar la integridad de la información almacenada. También se utilizó para preparar datos de prueba y hacer limpieza posterior a las pruebas.
- vi. Google Meet. Plataforma de comunicación que se utilizó para realizar reuniones virtuales con los equipos de desarrollo, calidad y con el área usuaria de la organización. Facilitó la coordinación de pruebas, revisión de observaciones y toma de decisiones en tiempo real.
- vii. Internet Information Services (IIS). Servidor web empleado para el despliegue del FRONTEND, BACKEND y API GATEWAY en entornos no productivos. Fue fundamental para validar la correcta publicación de servicios y su disponibilidad para el FRONTEND y para Postman.
- viii. Plan de pruebas elaborado. Documento técnico diseñado por el bachiller, que incluyó los objetivos de prueba, el alcance, los criterios de aceptación, los tipos de pruebas a realizar, la estrategia de validación, y los casos de prueba definidos para cada funcionalidad del sistema. Este plan fue un instrumento clave para la ejecución estructurada y trazable del proceso de pruebas.
- ix. Checklist de cumplimiento de estándares. Instrumento documental desarrollado y utilizado para asegurar que las actividades de control de calidad cumplieran con los estándares internos de Caja Huancayo y las buenas prácticas reconocidas a nivel internacional. Este *checklist* fue aplicado en cada fase del proceso de pruebas y se convirtió en una guía esencial para mantener la trazabilidad, consistencia y conformidad con las normativas exigidas por la organización. Este documento se alinea con:
 - Los principios del ISTQB (International Software Testing Qualifications Board), específicamente en cuanto a tipos de prueba, documentación y gestión de defectos.

- La norma ISO/IEC/IEEE 29119, que establece procesos, documentación, técnicas y métricas formales para la gestión de pruebas de *software*.

Estos instrumentos combinados permitieron al autor desarrollar sus actividades de control de calidad con eficiencia, precisión y alineación a los estándares del proyecto, asegurando la entrega de un sistema confiable, funcional y apto para su uso por parte del público objetivo.

A continuación, se detalla el plan de pruebas elaborado para el proyecto descrito en el presente caso práctico:

N.º	Caso de uso	Escenario	Entrada/acción	Resultado esperado	Tipo
1	CU01-Registro de clave web nueva	Ideal	Tarjeta válida + DNI correcto + clave válida	Clave registrada exitosamente. Usuario redirigido a inicio de sesión.	Positivo
		No ideal-Tarjeta inválida	Número de tarjeta no asociado a una Cuenta Futuro	Mensaje: “La tarjeta ingresada no es válida o no está asociada a una Cuenta Futuro activa”.	Negativo
		No ideal-Clave débil	Clave de 3 caracteres o sin combinación de letras y números	Mensaje: “La clave debe tener al menos 6 caracteres y contener letras y números”.	Negativo
2	CU02-Inicio de sesión	Ideal	Ingreso correcto de clave web	Usuario accede correctamente a la plataforma.	Positivo
		No ideal-Clave incorrecta	Ingreso de clave incorrecta	Mensaje: “Clave incorrecta. Inténtelo nuevamente”.	Negativo
		No ideal-Campos vacíos	No ingresar datos	Mensaje: “Debe ingresar su clave web para continuar”.	Negativo
3	CU03-Validación de tarjeta	Ideal	Tarjeta activa de Cuenta Futuro ingresada	Validación exitosa. Se continúa con el proceso de acceso o registro.	Positivo
		No ideal-Tarjeta inactiva	Tarjeta bloqueada o sin vínculo a Cuenta Futuro	Mensaje: “Tarjeta no válida o cuenta inactiva”.	Negativo
4	CU04-Recuperación de clave web	Ideal	Verificación correcta vía código de validación	Clave restablecida y mensaje de confirmación mostrado.	Positivo
		No ideal-Código incorrecto	Ingreso de código vencido o erróneo	Mensaje: “Código de verificación inválido o expirado”.	Negativo
5	CU05-Selección de personaje	Ideal	Selección válida de un personaje	Personaje asignado correctamente al perfil del usuario.	Positivo
		No ideal-No selecciona nada	Usuario no elige personaje y continúa	Mensaje: “Debe seleccionar un personaje para continuar”.	Negativo

6	CU06 - Selección de habilidad para el personaje	Ideal	Selección de habilidad	Habilidad seleccionada correctamente al personaje elegido	Positivo
7	CU07-Cargar juegos	Ideal	Usuario accede a sección de juegos	Juegos cargan correctamente y son interactivos	Positivo
		No ideal-Fallo en carga	Fallo en conexión o en acceso a la sección	Mensaje: “No se pudieron cargar los juegos. Intente más tarde”.	Negativo
8	CU08 - Ver datos de la cuenta	Ideal	Selección la opción para ver el detalle de la cuenta	Los datos mostrados correspondan a la cuenta y al usuario autenticado.	Positivo
9	CU09 - Editar personaje	Ideal	Edición de personaje seleccionado anteriormente	Personaje asignado correctamente al perfil del usuario.	Positivo
10	CU10 - Editar habilidad	Ideal	Edición de habilidad seleccionado anteriormente	Habilidad seleccionada correctamente al personaje elegido	Positivo
11	CU11-Registrar meta de ahorro	Ideal	Monto, motivo y plazo válidos ingresados	Meta de ahorro registrada exitosamente.	Positivo
		No ideal-Campos vacíos	Usuario deja en blanco algún campo obligatorio	Mensaje: “Debe completar todos los campos para registrar una meta”.	Negativo
		No ideal-Monto negativo	Usuario ingresa monto con valor negativo o 0	Mensaje: “El monto debe ser mayor a cero”.	Negativo
12	CU12-Ver saldo y movimientos	Ideal	Usuario autenticado accede al módulo de consulta	Se muestra saldo actual, fecha y lista de movimientos correctamente.	Positivo
		No ideal-Error de sesión	Sesión expirada o no autenticada	Mensaje: “Su sesión ha expirado. Por favor, inicie sesión nuevamente”.	Negativo
13	CU13-Ganar medalla	Ideal	Usuario cumple meta de ahorro dentro del plazo establecido	Sistema asigna medalla, muestra notificación o mensaje de logro.	Positivo

		No ideal- Meta no cumplida	Usuario no cumple meta antes del plazo	No se otorga medalla. Se muestra mensaje informativo.	Negativo
14	CU14-Cambiar de cuenta	Ideal	Usuario tiene múltiples Cuentas Futuro y selecciona otra cuenta	La plataforma cambia correctamente la información mostrada sin cerrar sesión	Positivo
		No ideal- Cuenta no disponible	Selección de cuenta no activa o sin autorización	Mensaje: “La cuenta seleccionada no está disponible”.	Negativo
15	CU15-Cerrar sesión	Ideal	Usuario cierra sesión desde menú o botón correspondiente	Usuario es redirigido a la página de inicio de sesión.	Positivo
		No ideal- Sesión ya expirada	Usuario intenta cerrar sesión después de haber sido desconectado por inactividad	Mensaje: “La sesión ya fue cerrada automáticamente por inactividad”.	Negativo

Figura 7. Plan de pruebas del proyecto plataforma Web Cuenta Futuro

Fuente: Elaboración propia

4.2.4. Equipos y materiales utilizados en el desarrollo de las actividades

- Equipo con sistema operativo Windows 11.
- Acceso a navegadores web (Chrome, Firefox, Edge) para pruebas cruzadas.
- Acceso al entorno de pruebas.
- Uso de tarjetas no productivas para la ejecución de pruebas.
- Acceso a servidores no productivos donde se realizarán las publicaciones.
- Acceso a la arquitectura productiva para la réplica en el entorno de pruebas.
- Manuales y documentación técnica del producto “Cuenta Futuro”.
- Acceso a bases de datos de prueba.
- Acceso a herramientas de ejecución de código, de base de datos y pruebas de API (Visual Studio, Visual Code, SQL Server Management, Postman).

4.3. Ejecución de las Actividades Profesionales

4.3.1. Cronograma de actividades realizadas

A continuación, se detalla por semanas el cronograma de actividades realizadas donde intervienen las actividades profesionales del analista de control de calidad de sistemas par aun proyecto como el descrito en el presente capítulo:

Cronograma de ejecución de pruebas de <i>software</i>		
Semana	Actividad a realizar	Responsable (s)
1	Revisión de requerimientos funcionales y técnicos del sistema	Analista de Control de Calidad de Software
2	Diseño del plan de pruebas, <i>checklist</i> de estándares y elaboración de casos de prueba	Analista de Control de Calidad de Software
3-4	Preparar el entorno de pruebas y habilitación de data para las pruebas (tarjetas, cuentas, clientes)	Analista de Control de Calidad de Software
5-6	Ejecución de pruebas funcionales (caja negra), exploratorias y validación de interfaz (FRONTEND/BACKEND)	Analista de Control de Calidad de Software
7-8	Pruebas de API con Postman y publicación de servicios en IIS	Analista de Control de Calidad de Software
9	Pruebas de integración y pruebas de caja blanca	Analista de Control de Calidad de Software
10	Pruebas de regresión y validación con el área usuaria mediante reuniones virtuales y presenciales	Analista de Control de Calidad de Software Área usuaria
11	Elaboración del informe de validación final, entrega de evidencia de pruebas y aplicación del <i>checklist</i> de estándares	Analista de Control de Calidad de Software
12-15	Envío a despliegue, soporte en producción y cierre del ciclo de pruebas, monitoreo de comportamiento dentro de la etapa de piloto del sistema	Equipo de solución Equipo de Producción Equipo de servidores área usuaria
16-19	Informe sobre estabilización en la etapa de piloto, indicación para el despliegue masivo y seguimiento del funcionamiento del sistema posterior al despliegue masivo	Equipo de solución
20	Documentación y envío del manual técnico del nuevo sistema, manual de configuración para el ambiente no productivo e informe final sobre estabilización del sistema	Equipo de solución

Figura 8. Cronograma de actividades profesionales

Fuente: Elaboración propia

4.3.2. Proceso y secuencia operativa de las actividades profesionales

El desarrollo de las actividades profesionales del bachiller como analista de Control de Calidad de Software se realizó siguiendo una secuencia lógica y estructurada. A continuación, se detallan las etapas del proceso y su secuencia operativa:

- i. Aterrizaje de requerimientos. El proceso se inicia con el análisis de requerimientos funcionales y no funcionales, liderado por el analista funcional, con participación del equipo de desarrolladores y analistas de control de calidad para entender el alcance del sistema, definir criterios de validación, identificar falencias y brindar más detalle de casos no especificados.
- ii. Revisión de documentación técnica y funcional. El analista revisa a detalle la documentación proporcionada, verificando la claridad de los requerimientos y anticipando posibles riesgos de calidad.
- iii. Diseño del plan de pruebas. Se elabora el plan de pruebas de acuerdo con las historias de acuerdo con casos de uso identificados, incluyendo la estrategia, los criterios de aceptación, el alcance, tipos de pruebas y herramientas a emplear.
- iv. Preparación del ambiente no productivo y habilitación de data para pruebas. Antes de comenzar las pruebas, se prepara un ambiente de pruebas que replicará las condiciones del entorno de producción, asegurando que no afectará las operaciones reales. Además, se habilitan datos específicos para realizar las pruebas, como tarjetas, cuentas y clientes ficticios, de manera que se pudieran ejecutar pruebas sin comprometer la privacidad ni la seguridad de datos reales.
- v. Diseño y ejecución de casos de prueba. Se diseñan y ejecutan pruebas de caja negra, caja blanca y pruebas exploratorias, abarcando los principales módulos del sistema. Se documentan los resultados y se identifican errores.
- vi. Informe de avance mediante reuniones diarias y presentaciones semanales. A lo largo del proceso de validación, el equipo de solución informa y presenta avances a través de *daily meetings* y *reuniones semanales* de seguimiento con *stakeholders*. Esto permite identificar obstáculos, priorizar pruebas críticas y alinear expectativas con el área usuaria.
- vii. Publicación de proyectos en servidores no productivos: Una vez superadas las pruebas unitarias en entorno local, los desarrollos se despliegan en servidores no productivos (ambiente de pruebas), utilizando Internet Information Services (IIS) para simular condiciones reales de operación y facilitar la interacción entre módulos del sistema.
- viii. Pruebas de integración entre proyectos: Se realizan pruebas para validar la correcta interacción entre los distintos componentes del sistema (FRONTEND, BACKEND, bases de

datos y API GATEWAY). Estas pruebas permiten verificar que la comunicación entre módulos se realiza de forma coherente, estable y segura.

- ix. Reporte y seguimiento de errores. Los defectos encontrados se registran en Azure DevOps como BUGS. El analista de control de calidad realiza un seguimiento exhaustivo hasta la corrección y validación de cada uno de ellos.
- x. Envío a pruebas de aceptación por el usuario. Una vez estabilizado el sistema en el entorno de pruebas, se procede a liberar una versión controlada para ser validada por el área usuaria de Caja Huancayo, quienes pueden verificar de esta manera que la plataforma cumple con sus expectativas funcionales y operativas.
- xi. Seguimiento a pruebas de usuario. El analista de control de calidad acompaña el proceso de pruebas de usuario mediante reuniones virtuales, soporte directo y levantamiento de observaciones, asegurando que la retroalimentación sea documentada y se dé seguimiento para su atención oportuna.
- xii. Aplicación del *checklist* de estándares. Se aplica el *checklist* de cumplimiento basado en ISTQB e ISO/IEC/IEEE 29119, asegurando que las pruebas cumplen con estándares internacionales y políticas internas de calidad.
- xiii. Informe de validación final. Se consolidan los resultados obtenidos, se documentan los defectos solucionados, se registran las observaciones atendidas y se presenta un informe de validación final para aprobación del sistema.
- xiv. Soporte en producción y cierre de pruebas. Durante el despliegue, el bachiller participa en la verificación del comportamiento del sistema en ambiente productivo, asegura la transición fluida y sin incidentes. Se procede luego al cierre formal del proceso de pruebas.

5. CAPÍTULO V

RESULTADOS

Este capítulo presenta los resultados obtenidos a partir del ejercicio profesional desempeñado por el bachiller en el cargo de analista de Control de Calidad de Software, en el contexto de implementación de una página web dirigida a menores de edad, cuyo objetivo principal es fomentar la cultura del ahorro a través del uso del producto financiero “Cuenta Futuro” de la Caja Huancayo. El enfoque se centra en la validación de la calidad del *software* y cómo este proceso asegura la funcionalidad, usabilidad y confiabilidad del producto final, a fin de garantizar una experiencia adecuada para su público objetivo.

5.1. Resultados Finales de las Actividades Realizadas

En relación con el primer objetivo específico, que buscaba comprobar la alineación del proceso de pruebas en el Área de Control de Calidad de Sistemas de la Caja Huancayo con las normas internacionales ISO/IEC/IEEE 29119 y el estándar ISTQB, se logra evidenciar un cumplimiento adecuado en todas las etapas del ciclo de vida del *software*. Durante la ejecución de las pruebas funcionales, de interfaz, rendimiento e integración, se identificaron y documentaron aproximadamente 65 incidentes, las cuales fueron corregidas antes del pase a producción, logrando así una reducción significativa de los costos asociados a ajustes posproductivos. Las pruebas manuales se llevaron a cabo de manera exhaustiva, enfocándose en los módulos de mayor impacto y validando interacciones críticas, tales como el acceso al sistema, siguiendo las mejores prácticas recomendadas por ISTQB. Además, se implementó exitosamente un entorno de preproducción donde se realizó un piloto con usuarios reales bajo condiciones supervisadas, sin que se reportaran fallas críticas. Esta validación anticipada facilitó un despliegue efectivo en ambiente productivo, a fin de garantizar la correcta operación de todos los servicios y *endpoints* de la plataforma, sin interrupciones ni errores de integración con los sistemas internos de la Caja Huancayo. Finalmente, la ausencia de observaciones relevantes durante las primeras semanas tras el lanzamiento reafirmó la efectividad del proceso de aseguramiento de calidad implementado.

En concordancia con el segundo objetivo específico, que planteaba la documentación de la aplicación del sistema mediante la creación de historias de usuario, planes de prueba y casos de prueba, se desarrolla una validación integral de los requisitos funcionales y no funcionales del sistema. Esta validación se sustentó en el diseño y ejecución de casos de prueba elaborados a partir de las historias de usuario y escenarios definidos juntamente con el equipo de negocio. Asimismo, se generó documentación técnica completa que incluyó la descripción de los casos de prueba, los resultados obtenidos, las incidencias reportadas y la evidencia de las ejecuciones realizadas. Todo este proceso evidenció la aplicación práctica del conocimiento adquirido durante la carrera de Ingeniería de Sistemas

e Informática, a través del uso de herramientas especializadas y metodologías de aseguramiento de calidad estructuradas y estandarizadas a lo largo del proyecto.

Respecto al tercer objetivo específico, orientado a la identificación de factores internos que afectaron o retrasaron la ejecución de las labores en el Área de Control de Calidad, se destacó la importancia del entorno de pruebas no productivo como un espacio clave para la detección temprana de riesgos, lo que facilitó la mitigación oportuna de problemas antes del despliegue en producción. Durante el proceso, se identificaron ciertas limitaciones logísticas y de recursos que representaron desafíos para el equipo, pero que también permitieron implementar acciones preventivas que evitaron retrasos mayores y contribuyeron a la optimización de los procesos de control de calidad.

En conjunto, estos resultados finales reflejan un impacto positivo y directo del trabajo realizado en el aseguramiento de la calidad del *software*, a fin de garantizar una solución tecnológica robusta, funcional y segura, especialmente diseñada para un público sensible como lo son los menores de edad, quienes requieren una experiencia intuitiva y confiable.

5.2. Logros Alcanzados

Como resultado del ejercicio profesional realizado en el Área de Control de Calidad de Sistemas de la Caja Huancayo, se alcanzaron logros significativos que reflejan la efectividad de las actividades desarrolladas y la aplicación del conocimiento adquirido en la carrera de Ingeniería de Sistemas e Informática. Entre los logros más relevantes destaca un incremento del 95 % en el cumplimiento de los criterios de aceptación del producto, lo que facilitó que el área de desarrollo entregue una solución más estable y alineada con las necesidades y expectativas del cliente. Este avance tuvo un impacto directo en la calidad del *software*, asegurando la entrega de un producto robusto y funcional.

Asimismo, se evidenció una disminución significativa de errores en producción, manifestada en una reducción considerable en la cantidad de reportes de soporte técnico durante las primeras semanas posteriores al lanzamiento inicial de la plataforma. Este logro confirma la efectividad del proceso de aseguramiento de calidad implementado y su contribución a la mejora continua del producto.

Además, se logró la estandarización de los procesos de control de calidad mediante la adopción de metodologías ágiles, como Scrum, y la aplicación de técnicas de pruebas fundamentadas en criterios reconocidos internacionalmente, tales como los establecidos por ISTQB. Esta estandarización permitió optimizar el flujo de trabajo, mejorar la comunicación entre equipos y garantizar la consistencia en la ejecución de las pruebas.

Finalmente, se cumplió de manera rigurosa con las actividades planificadas según el cronograma establecido, respetando los plazos definidos en cada sprint y entregando los entregables de aseguramiento de calidad en las fechas previstas. Este cumplimiento puntual contribuyó a la eficiencia

general del proyecto y fortaleció la confianza del equipo de desarrollo y de los *stakeholders* en el área de control de calidad.

Estos logros demuestran que la intervención realizada no solo permitió alcanzar los objetivos planteados, sino que también sentó las bases para la mejora continua del proceso de aseguramiento de calidad en la Caja Huancayo.

5.3. Dificultades Encontradas

Durante el desarrollo del proceso de aseguramiento de calidad de la plataforma web educativa de Caja Huancayo, se presentaron diversas dificultades que afectaron el ritmo y la eficiencia de las actividades planificadas. Estos impedimentos fueron abordados mediante acciones correctivas y lecciones aprendidas para futuros proyectos similares. A continuación, se detallan las principales:

- i. Falta de claridad en los requisitos de algunos casos de uso, lo que generó múltiples reformulaciones en los casos de prueba y retrasos en la validación de funcionalidades clave.
- ii. Limitaciones en el entorno de pruebas, particularmente en la disponibilidad de dispositivos móviles y navegadores utilizados comúnmente por los usuarios menores de edad, dificultando una validación completa del comportamiento multiplataforma en la visualización de interfaces.
- iii. Restricciones de tiempo en los ciclos de pruebas, debido a un cronograma ajustado que exigió la priorización de pruebas de alto impacto, dejando algunas funcionalidades secundarias con menor cobertura.
- iv. Lentitud en la ejecución de pruebas manuales, ya que este tipo de pruebas implicó una mayor inversión de tiempo y recursos humanos para validar regresiones y flujos repetitivos.
- v. Lentitud inherente a la metodología de desarrollo en cascada, que no permitió una retroalimentación temprana ni la validación progresiva de entregables, postergando la identificación de errores hasta etapas avanzadas del proyecto.
- vi. Número considerable de observaciones encontradas durante la fase de *testing*, especialmente relacionadas con la validación de reglas de negocio, errores de interfaz gráfica y validaciones incorrectas, lo que requirió una revisión exhaustiva de varios módulos antes del pase a producción y reprocesos en la ejecución de pruebas. Estas dificultades no solo revelaron la necesidad de fortalecer el enfoque metodológico y técnico del proceso de desarrollo, sino también la importancia del rol del analista de calidad como un facilitador clave para garantizar un producto funcional, confiable y centrado en el usuario final.

5.4. Planteamiento de Mejoras

Con base en las dificultades detectadas durante la implementación de la plataforma web educativa para menores de edad, se proponen una serie de mejoras orientadas a optimizar los procesos

de control de calidad, reducir tiempos de respuesta y fortalecer la entrega de productos con mayor valor para la organización. Estas mejoras se dividen en metodologías propuestas y la descripción de su implementación.

5.4.1. Metodologías propuestas

Para elevar la eficiencia del proceso de aseguramiento de calidad y reducir las limitaciones observadas, se proponen las siguientes metodologías y prácticas:

- i. Uso de pruebas automatizadas. Se recomienda implementar pruebas automatizadas utilizando herramientas como Selenium para pruebas de interfaz, Newman para servicios REST y JUnit/TestNG en pruebas unitarias, lo que permitirá validar funcionalidades recurrentes de forma más rápida, reducir errores humanos y agilizar las pruebas de regresión.
- ii. Adopción de metodologías ágiles (Scrum). Se sugiere cambiar el enfoque tradicional en cascada por metodologías ágiles que permitan una mejor planificación y ejecución incremental. En especial, se promoverá la disgregación de historias de usuario en entregables mínimos viables (MVPs), facilitando ciclos cortos de desarrollo, validación temprana y entrega continua de valor al cliente.
- iii. Aplicación de pruebas de exploración. Además de las pruebas estructuradas, se integrarán sesiones de pruebas exploratorias por parte del equipo QA, orientadas a validar aspectos de usabilidad e interacción en entornos controlados con usuarios reales (niños y padres) y descubrir fallos no previstos en los casos de prueba evaluando la experiencia de usuario.
- iv. Optimización del seguimiento posproducción. Se plantea acortar el periodo de monitoreo intensivo del equipo de solución una vez que el sistema entra en producción. Esto busca reducir el impacto sobre otras actividades asignadas al equipo técnico, sin comprometer la calidad, mediante la implementación de alertas automatizadas, monitoreo proactivo y una fase de estabilización más efectiva antes del despliegue final.

5.4.2. Descripción de la implementación

Para llevar a cabo estas mejoras, se propone un plan dividido en fases:

- i. Capacitación y alineamiento del equipo técnico. Introducción de herramientas y prácticas de pruebas automatizadas, así como formación en metodologías ágiles y redacción efectiva de historias de usuario orientadas a MVPs.
- ii. Redefinición del flujo de trabajo del equipo QA y desarrollo. Reorganización de tareas bajo un enfoque Scrum, con *sprints* de duración fija, definición de criterios de aceptación claros y revisiones incrementales.

- iii. Implementación piloto de pruebas automatizadas. Iniciar con la automatización de funcionalidades críticas y de alta frecuencia, y establecer una integración con el *pipeline* de despliegue.
- iv. Inclusión de sesiones de pruebas exploratorias. Programar actividades de prueba libre por parte del equipo QA al cierre de cada sprint, especialmente enfocadas en aspectos de usabilidad e interacción general del sistema y considerar el impacto de incluir usuarios externos en las pruebas de aceptación
- v. Reestructuración del seguimiento posproducción. Reducir la dependencia del equipo de solución en la etapa posproducción mediante mejores prácticas de despliegue, monitoreo en tiempo real y soporte de primer nivel más autónomo.

Estas acciones están diseñadas para mejorar tanto la eficiencia del proceso de pruebas como la calidad percibida del producto final, reduciendo la carga operativa posterior y alineando el desarrollo de *software* a las necesidades reales del negocio y los usuarios.

5.5. Análisis

El análisis de los resultados demuestra que el trabajo de control de calidad no solo contribuye a la reducción de errores, sino también a los siguientes:

- Incrementar la satisfacción del usuario final, al ofrecer un producto funcional, amigable y estable.
- Reducir los costos de mantenimiento, al detectar errores antes de la etapa de producción.
- Aumentar la eficiencia del equipo de desarrollo, al contar con retroalimentación oportuna y casos de prueba bien definidos.

En particular, en el caso de la presente plataforma web, el análisis reveló que más del 85 % de los usuarios pudieron navegar correctamente por el sitio y cumplir las tareas de simulación de ahorro, lo cual valida su funcionalidad y pertinencia pedagógica.

5.6. Aporte del Bachiller en la Empresa y/o Institución

El aporte del bachiller a Caja Huancayo se traduce en lo siguiente:

- Fortalecimiento del proceso de calidad de *software*, lo que elevó los estándares de validación aplicados en proyectos internos.
- Generación de documentación estandarizada, incluyendo matrices de pruebas, informes de errores y métricas de cobertura.

- Generación de manuales de configuración no productiva para el levantamiento de ambientes en entornos de pruebas en futuros requerimientos de mantenimiento u optimización de procesos.
- Propuesta de automatización para la optimizando recursos y tiempos de entrega.
- Desempeño como facilitador entre el equipo de desarrollo, negocio y usuarios, promoviendo la mejora continua y alineación de expectativas.

En síntesis, el trabajo profesional desempeñado ha impactado positivamente en la calidad de los productos tecnológicos de la empresa, mejorando la experiencia del cliente y reforzando la confianza de los usuarios en las soluciones digitales de Caja Huancayo.

CONCLUSIONES

Este proyecto de suficiencia profesional ha permitido demostrar de forma significativa la aplicación práctica de los conocimientos adquiridos durante la carrera de Ingeniería de Sistemas e Informática en el desempeño profesional como analista de Control de Calidad de Software en la Caja Municipal de Ahorro y Crédito Huancayo. Mediante el análisis y la documentación de un caso real centrado en el desarrollo e implementación de una plataforma web dirigida a menores de edad, vinculada al producto financiero “Cuenta Futuro”, se ha evidenciado que las habilidades y competencias obtenidas en la formación académica son no solo relevantes sino indispensables para un desempeño eficaz, conforme a los estándares requeridos por el sector financiero.

En primer término, se ha comprobado que las actividades en el área de Control de Calidad no son simples tareas operativas desconectadas, sino procesos estratégicos integrados dentro del ciclo de vida del *software*, que demandan un profundo conocimiento en ingeniería de *software*, gestión de pruebas y normativas internacionales. En este contexto, se verificó que el proceso de pruebas aplicado en la organización está alineado con la norma ISO/IEC/IEEE 29119 y con las buenas prácticas promovidas por el International Software Testing Qualifications Board (ISTQB), estableciendo un marco metodológico sólido para el diseño, ejecución y supervisión de la calidad del *software* desarrollado. Este alineamiento garantiza no solo la trazabilidad y formalidad del proceso, sino también su mejora continua, beneficiando tanto al usuario final como a la entidad.

Asimismo, se constató que la planificación y ejecución de pruebas están estrechamente ligadas al conocimiento técnico y metodológico adquirido durante la formación universitaria. La identificación de funcionalidades críticas a partir del análisis del sistema y la elaboración de historias de usuario sirvieron como base para construir planes de prueba detallados, que incluyeron escenarios y casos de prueba para validar el funcionamiento del sistema tanto en condiciones ideales como en situaciones de riesgo. Esta integración de teoría y práctica permitió que el producto final cumpliera con los requisitos funcionales y no funcionales definidos, alcanzando los estándares de calidad esperados.

Por otra parte, el desarrollo del proyecto permitió identificar varios factores internos que afectan la eficiencia del área de control de calidad. Entre ellos destaca la falta de herramientas de automatización adecuadas para pruebas repetitivas, cuya implementación podría optimizar significativamente los tiempos de atención y elevar la calidad del producto, al permitir que el analista se concentre en revisiones más detalladas y de mayor impacto. Además, la creciente carga de trabajo relacionada con el monitoreo postproducción, especialmente durante períodos prolongados necesarios para asegurar la estabilidad del sistema, reduce la capacidad del analista para atender otras tareas planificadas dentro del ciclo de pruebas. Esta situación representa un desafío en la gestión del tiempo y la priorización, dado que algunos proyectos requieren seguimiento por meses o incluso años, lo que limita la capacidad de respuesta frente a nuevas solicitudes del negocio y aumenta la cantidad de proyectos bajo supervisión. Aunque estos

aspectos no impiden el cumplimiento de los objetivos del área, constituyen limitaciones que deben ser abordadas estratégicamente para mejorar la eficiencia y efectividad del aseguramiento de calidad. Reconocer estas dificultades también abre la puerta a propuestas de mejora que fortalezcan el área a mediano y largo plazo.

Finalmente, se concluye que el rol del analista de Control de Calidad es fundamental dentro del ciclo de vida del desarrollo de *software*, especialmente en entidades financieras donde la precisión, seguridad y confiabilidad son esenciales. El trabajo realizado evidencia que la calidad del producto final es el resultado de un proceso riguroso que incluye análisis, planificación, validación y mejora continua. La experiencia profesional recogida en este documento ejemplifica cómo la formación en ingeniería de sistemas e informática se traduce en una práctica profesional competente, ética y orientada a generar valor en la organización.

RECOMENDACIONES

Basándose en los resultados obtenidos durante el desarrollo de este trabajo de suficiencia profesional, y tras analizar la experiencia laboral desempeñada como analista de Control de Calidad de Software en la Caja Huancayo, se plantean a continuación una serie de recomendaciones dirigidas a mejorar el proceso de aseguramiento de calidad, optimizar el uso de recursos en el área y consolidar las mejores prácticas en el desarrollo de *software* dentro del ámbito institucional, tomando como referencia las oportunidades de mejora detectadas.

En primer lugar, se sugiere fortalecer la integración del área de calidad desde las fases iniciales del ciclo de vida del *software*, fomentando una participación activa en la definición de requisitos, la revisión anticipada de historias de usuario y la identificación temprana de requerimientos esenciales para una adecuada validación en ambientes no productivos. Esta estrategia permitirá detectar defectos desde etapas tempranas, ampliar la cobertura de pruebas y disminuir los costos derivados de correcciones realizadas en fases posteriores. Para ello, se recomienda implementar espacios de trabajo colaborativo entre analistas de calidad, desarrolladores y analistas funcionales, promoviendo la mejora continua.

De igual manera, se aconseja avanzar hacia la adopción completa de metodologías ágiles para la gestión de proyectos de desarrollo de *software*, particularmente en iniciativas orientadas a la innovación de productos digitales y nuevas funcionalidades. La implementación de metodologías ágiles, como Scrum, facilitaría la entrega de productos en versiones mínimas funcionales, que pueden ser validadas y perfeccionadas de forma progresiva. Este enfoque favorecería una mayor flexibilidad ante cambios en los requisitos, una retroalimentación más rápida del usuario final y una coordinación más eficiente entre los distintos roles del equipo de desarrollo, incluyendo al área de calidad como un componente clave en cada sprint o iteración.

En términos operativos, se recomienda evaluar la incorporación gradual de herramientas de automatización que permitan disminuir la carga asociada a pruebas repetitivas, liberando tiempo para que el personal se concentre en actividades que aporten un mayor valor analítico.

Respecto a los factores internos detectados, es vital establecer mecanismos que permitan priorizar y balancear la carga de trabajo, minimizando el impacto que las tareas de monitoreo y seguimiento postproducción tienen sobre la planificación general del área. Se aconseja definir límites temporales razonables para estas actividades, así como criterios claros para su escalamiento y cierre, con el fin de evitar que interfieran en la atención de nuevas funcionalidades o proyectos. Esta acción requiere una coordinación efectiva con las áreas de soporte y desarrollo para asegurar una distribución adecuada de responsabilidades y una visión común del proceso de aseguramiento de calidad.

Por último, se recomienda fomentar una cultura organizacional enfocada en la calidad, que reconozca el valor estratégico del Control de Calidad de Software dentro de la institución. Para ello, se

propone promover espacios de formación continua, certificaciones profesionales como ISTQB para el personal del área, y la implementación de indicadores de desempeño que evidencien el impacto del trabajo realizado. Estas iniciativas no solo contribuirán al fortalecimiento técnico del equipo, sino también al reconocimiento institucional del rol fundamental que desempeña en la generación de productos de *software* confiables, seguros y alineados con los objetivos de la Caja Huancayo.

BIBLIOGRAFÍA

1. GARCÍA, J. La Caja Municipal de Ahorro y Crédito de Huancayo: Un estudio sobre el caso sobre impacto social en la banca peruana. *Revista de Economía y Finanzas*, 2019, 15(2), p. 45-58. ISSN 2600-8128.
2. SÁNCHEZ, M. Responsabilidad Social Corporativa en la Caja Huancayo: Un análisis de sus programas y prácticas. *Investigación Económica*, 2020, 25(3), p. 112-125. ISSN 2790-592X.
3. LINARES, M. *Informe de Procesos de Caja Huancayo y Propuesta de Mejora*. Agosto 2019. Disponible en: <https://es.slideshare.net/mlinaresvizcarra/informe-caja-huancayo>. Consultado el: 21 may. 2024.
4. CAJA MUNICIPAL DE AHORRO Y CRÉDITO DE HUANCAYO. *Misión y Visión*. Huancayo: Caja Municipal de Ahorro y Crédito de Huancayo, 2016. Disponible en: https://www.cajahuancayo.com.pe/PCM_NuesCaja/PCM_frmMisVision.aspx?cCodigo=5. Consultado el: 21 may. 2024.
5. IDROVO, F. E. *Diseño de Manual de Procedimientos de Fase de Pruebas para Mejorar la Calidad del Software*. Trabajo de Titulación Especial (Grado de Magister en Administración de Empresas). Guayaquil, Ecuador: Universidad de Guayaquil, 2016.
6. DUQUINO, Á. P. *Automatización de un Sistema de Pruebas de Software para la Optimización del Proceso de Calidad de DetectID*. Trabajo de Grado (Grado de Magister en Automatización Industrial). Bogotá, Colombia: Universidad Nacional de Colombia, 2020.
7. ROMÁN, F. M. *Procedimiento Metodológico de Pruebas de Software para Optimizar el Control de Avance de los Proyectos de Testing en la Empresa Tsoft*. Tesis (Título profesional de Ingeniería de Sistemas). Lima, Perú: Universidad Femenina del Sagrado Corazón, 2016.
8. SCHWABER, K. y SUTHERLAND, J. *La Guía Scrum*. Traducido por David Hernán Tardini y David Marti, 2020. Disponible en: <https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-Spanish-Latin-South-American.pdf>. Consultado el: 21 may. 2024.
9. ORGANISMO DE NORMALIZACIÓN EN ESPAÑA. *ISO/IEC/IEEE 29119 El nuevo estándar internacional para pruebas de software*. San Sebastián: AENOR, 2015.
10. INTERNATIONAL SOFTWARE TESTING QUALIFICATIONS BOARD (ISTQB). *What we do*. 2018. Disponible en: <https://www.istqb.org/>. Consultado el: 21 may. 2024.
11. BALAREZO, J. C. *Aplicación de las Buenas Prácticas Basadas en el ISTQB para la Gestión del Área de Calidad de Software*. Trabajo de Suficiencia Profesional (Título Profesional de Ingeniería Empresarial y de Sistemas). Lima, Perú: Universidad San Ignacio de Loyola, 2022.

12. REALPE, L. C. *Metodología para el proceso de pruebas de software: Un estudio de caso enfocado a una empresa de software colombiana*. Tesis (Grado de Magister en Ingeniería de Sistemas y Computación). Bogotá, Colombia: Universidad Nacional de Colombia, 2019.
13. CAJA MUNICIPAL DE AHORRO Y CRÉDITO DE HUANCAYO. *Cuenta Futuro*. Huancayo: Caja Municipal de Ahorro y Crédito de Huancayo, [s.f.]. Disponible en: https://www.cajahuancayo.com.pe/PCM_ProdServicios/PCM_frmSubProAho.aspx?id=SA002&cCodigo=31&cTipo=0. Consultado el: 21 may. 2024.

ANEXOS

Anexo 1. Fotografías

En este anexo se incluyen imágenes vinculadas a las actividades desempeñadas por el analista de control de calidad, las cuales muestran claramente el esfuerzo y la colaboración necesarios para ejecutar las tareas dentro del ciclo de vida del *software*.



Fotografía 1: Participación de Sprint Review en conjunto con el área de negocios



Fotografías 2 y 3: Participación de las reuniones diarias de seguimiento