Universidad Peruana de Ciencias Aplicadas

Facultad de Ingeniería

PROGRAMA ACADÉMICO DE INGENIERÍA DE SISTEMAS Y
COMPUTACIÓN

Modelo de certificación de pruebas no funcionales

TESIS

Para la obtención del Título Profesional de Ingeniero de Sistemas de
Información

Autores

Quezada Dávila, Manuel Alejandro (0000-0001-8629-5425)

Villarreal Agüero, Renzo Geraldo (0000-0001-6934-8943)

Asesor

Rivas Galloso, Paul (0000-0003-1879-4833)

Lima, 05 de Junio del 2018

A Dios y a nuestros padres.

Resumen

El presente proyecto profesional propone un Modelo de Certificación para el proceso de pruebas no funcionales de aplicaciones web o móviles basado en metodologías, buenas prácticas y estándares internacionales; el cual se encuentra compuesto de tres capas las cuales brindan soporte al proceso de testing de requerimientos no funcionales y una capa de soporte donde se definen las metodologías y marcos de trabajo que apoyaron al diseño del modelo.

Para efectos del desarrollo del proyecto, se ha investigado diversas fábricas de testing conjuntamente con los modelos de pruebas a los que se rigen y, asimismo, se identificó metodologías, estándares y buenas prácticas que ayudaron a construir el modelo a implementar. Cabe resaltar que el estándar internacional que tomó mayor relevancia para el diseño del modelo de certificación planteado fue la ISO 29119, el cual brinda un conjunto de buenas prácticas para ejecutar el proceso de testing de un producto software. Mediante la aplicación de un caso de estudio, se comprobó que el Modelo de Certificación de Pruebas no Funcionales reduce la complejidad del proceso de pruebas de los requerimientos no funcionales de aplicaciones web y móviles, optimiza los recursos para la ejecución de las pruebas, y permite identificar una mayor cantidad de deficiencias. Además, como resultado de un juicio de expertos, se concluyó que el Modelo de Certificación es fácil e intuitivo de implementar en cualquier organización.

Palabras clave: Certificación; Procesos; Estandarización; Aplicación web;

Tabla de contenido

Introducción.....	1
CAPITULO 1: DESCRIPCIÓN DEL PROYECTO	14
Objeto de Estudio	15
Dominio del Problema.....	15
Planteamiento de la Solución	15
Objetivos del Proyecto	16
Objetivo General.....	16
Objetivos Específicos	16
Indicadores de Éxito	16
Planificación del Proyecto	17
Alcance	17
Plan de Gestión del Tiempo	18
Plan de Gestión de Recursos Humanos	19
Plan de Comunicaciones.....	21
Plan de Gestión de Riesgos	22
CAPITULO 2: LOGROS DE LOS STUDENT OUTCOMES	24
Student Outcome A	25
Student Outcome B.....	25
Student Outcome C.....	26
Student Outcome D	27
Student Outcome E.....	28
Student Outcome F	29
Student Outcome G	30
Student Outcome H	31
Student Outcome I.....	32
Student Outcome J.....	32
Student Outcome K	33
Student Outcome J – CAC	34
WASC Student Outcome – Comunicación Escrita	34
WASC Student Outcome – Comunicación Oral	35
WASC Student Outcome – Ciudadanía.....	35
WASC Student Outcome – Pensamiento Innovador	36

WASC Student Outcome – Pensamiento Crítico	36
WASC Student Outcome – Razonamiento Cuantitativo	37
WASC Student Outcome – Manejo de la Información	37
CAPITULO 3: MARCO TEÓRICO	40
Certificación	41
Definición según AENOR	41
Definición según ISO	41
Definición según ISTQB	41
Síntesis.....	41
Tipos de Aplicaciones	42
Aplicaciones Web.....	42
Aplicaciones Móviles	42
Pruebas no Funcionales	44
Clasificación según SWEBOK.....	44
Clasificación según ISQTB	45
Clasificación según BURNSTEIN	47
Tipos de Pruebas por Ejecución	48
Pruebas Manuales	48
Pruebas Automatizadas.....	48
Técnicas de Diseño de Pruebas	50
Clasificación según ISTQB	50
Técnica de Pruebas Estáticas	51
Requerimientos de Software.....	52
Clasificación según SWEBOK.....	52
Clasificación según CMMI - DEV	52
Clasificación según ISQTB	53
Modelos Referenciales	53
V-MODEL, ISTQB	53
UML Testing Profile	55
ISO/IEEE 29119 – SOFTWARE TESTING STANDARD	56
CAPITULO 4: ESTADO DEL ARTE	60
Problema.....	61
Importancia del problema.....	61
Revisión de literatura.....	61

Motivación.....	63
Aporte	63
CAPITULO 5: DESARROLLO DEL PROYECTO	65
Análisis foda de fábricas de testing y software	66
Benchmarking de herramientas para pruebas no funcionales de aplicaciones web y móviles	66
Aplicaciones web.....	66
Aplicaciones móviles.....	76
Modelo de certificación de Pruebas No Funcionales	88
Gestión Organizacional del Proceso de Pruebas	88
Gestión del Proceso de Pruebas.....	88
Diseño y Caracterización del Modelo de Certificación de Pruebas No Funcionales .	89
CAPITULO 6: RESULTADO DEL PROYECTO.....	148
MODELO DE CERTIFICACIÓN DE PRUEBAS NO FUNCIONALES	149
Modelo de Certificación	149
Especificación de indicadores	149
Resultados de Evaluación del Modelo de Certificación.....	153
Análisis de factibilidad económica.....	156
Análisis	156
Costo de implementación	158
Análisis Financiero Final.....	160
CAPITULO 7: GESTIÓN DEL PROYECTO	162
Producto final	163
Gestión del Tiempo	164
Gestión de los Recursos Humanos	168
Gestión de las Comunicaciones	169
Gestión de los Riesgos.....	171
Lecciones aprendidas.....	173
Conclusiones.....	174
Recomendaciones	175
Glosario	176
Síglario	177

Índice de tablas

Tabla 1: Tabla de riesgos.....	22
Tabla 2: Clasificación de requerimientos no funcionales para app web	66
Tabla 3: Subcategoría de RNF Usabilidad App Web.....	67
Tabla 4: Subcategoría de RNF Movilidad app Web.....	68
Tabla 5: Subcategoría de RNF Performance app Web.....	68
Tabla 6: Subcategoría de RNF Seguridad app Web	68
Tabla 7: Subcategorías de RNF Escalabilidad app Web	70
Tabla 8: Subcategorías de RNF Monitoreabilidad app Web	71
Tabla 9: Herramientas de testing para app web.....	71
Tabla 10: Clasificación de requerimientos no funcionales para app móvil.....	76
Tabla 11: Subcategorías de RNF Usabilidad app móvil.....	78
Tabla 12: Subcategorías de RNF Conectividad app móvil.....	78
Tabla 13: Sub categorías de RNF Data app móvil	79
Tabla 14: Subcategorías de RNF Software app móvil	79
Tabla 15: Subcategorías de RNF interrupciones móviles app móvil	80
Tabla 16: Subcategorías de RNF Performance app móvil	80
Tabla 17: Subcategorías de RNF Seguridad app móvil.....	81
Tabla 18: Subcategorías de RNF Monitoreabilidad app móvil	82
Tabla 19: Herramientas de testing para app móviles.....	83
Tabla 20: Modelo de certificación de Pruebas No Funcionales	89
Tabla 21: Gestión Organizacional del Proceso de Pruebas	91
Tabla 22: Gestión del proceso de prueba.....	94
Tabla 23: Planificación.....	95
Tabla 24: Gestión de Recurso.....	98
Tabla 25 Control y seguimiento	101
Tabla 26 Gestión de Plan de Pruebas	103
Tabla 27: Definición de plan de pruebas	104
Tabla 28: Gestión de pruebas no funcionales.....	108
Tabla 29: Gestión de Pruebas No Funcionales para Aplicaciones Web.....	111
Tabla 30: Gestión de Pruebas No Funcionales para Aplicaciones Móviles	114
Tabla 31: Gestión ambiente de pruebas.....	117
Tabla 32: Monitoreo de pruebas.....	120
Tabla 33: Control y mejora de pruebas	122

Tabla 34: Reporte de pruebas	125
Tabla 35: Finalización	127
Tabla 36: Gestión de pruebas dinámicas	129
Tabla 37: Diseño e implementación de pruebas	130
Tabla 38 Ejecución de pruebas	133
Tabla 39: Gestión de informe de incidentes	135
Tabla 40 Cuadro de Evaluación de Indicador 1	150
Tabla 41 Cuadro de Evaluación de Indicador 2	150
Tabla 42 Cuadro de Evaluación de Indicador 3	151
Tabla 43 Cuadro de Evaluación de Indicador 4	151
Tabla 44 Cuadro de Cálculo de Factores Ambientales.....	152
Tabla 45 Cuadro de Cálculo de Factores Ambientales.....	153
Tabla 46: Matriz de Resultados	154
Tabla 47: Costos por fallas en ambiente de producción	157
Tabla 48: Pérdida Económica mensual	157
Tabla 49: Costo de horas	158
Tabla 50: Costo del Proyecto	159
Tabla 51: Costo de Herramientas para pruebas a app web.....	159
Tabla 52: Costo de Herramientas para pruebas a app mobiles.....	160
Tabla 53: Costo Total	160
Tabla 54: Beneficio de implementación	161
Tabla 55: Tabla de RRHH	169
Tabla 56: Tabla de Riesgos	172

Índice de Ilustraciones

Ilustración 1: Organigrama de empresas virtuales de UPC	19
Ilustración 2: Representación gráfica del Student Outcome C.....	27
Ilustración 3: Representación gráfica del Student Outcome D	28
Ilustración 4: Representación gráfica del Student Outcome E.....	28
Ilustración 5: Representación gráfica del Student Outcome G	31
Ilustración 6: V- Model	54
Ilustración 7: UML Testing Profile	56
Ilustración 8: Modelo de Referencia – ISO 29119	57
Ilustración 9: Herramientas seleccionadas con licencia para pruebas en web	75
Ilustración 10: Herramientas seleccionadas open source para pruebas en web	75
Ilustración 11: Herramientas seleccionadas con licencia para pruebas en web	87
Ilustración 12: Herramientas seleccionadas open source para pruebas en web	87
Ilustración 13: Modelo de certificación de Pruebas No Funcionales	89
Ilustración 14: Gestión organizacional del proceso de pruebas	91
Ilustración 15: Gestión del proceso de prueba	93
Ilustración 16: Planificación.....	95
Ilustración 17: Gestionar recurso.....	98
Ilustración 18: Control y Seguimiento	101
Ilustración 19: Gestión de plan de pruebas.....	102
Ilustración 20: Definición de plan de pruebas	104
Ilustración 21: Gestión Pruebas no funcionales	107
Ilustración 22: Gestión de Pruebas No Funcionales para Aplicaciones Web	110
Ilustración 23: Gestión de Pruebas No Funcionales para Aplicaciones Móviles.....	113
Ilustración 24: Gestión ambiente de pruebas.....	117
Ilustración 25: Monitoreo de pruebas	120
Ilustración 26: Control y mejora de pruebas	122
Ilustración 27: Reporte de pruebas	125
Ilustración 28: Finalización	127
Ilustración 29: Gestión de pruebas dinámicas	128
Ilustración 30: Diseño e implementación de pruebas.....	130
Ilustración 31: Ejecución de pruebas.....	132

Ilustración 32: Gestión de informe de incidentes	135
Ilustración 33: Gráfica de presentación de Resultados.....	155
Ilustración 34: Evaluación del Primer Sistema.....	155
Ilustración 35: Evaluación del Segundo Sistema	156
Ilustración 36: Cronograma de Proyecto	166
Ilustración 37: Árbol de entregables 1ra parte.....	167
Ilustración 38 Árbol de entregables 2da parte	168
Ilustración 39: Correo de IT-Services	170
Ilustración 40: Carpetas de Google Drive	170
Ilustración 41: Carpetas de Google Drive	171

Introducción

El presente documento se encuentra distribuido en 7 capítulos en donde se brinda detalles del planeamiento, ejecución, control y cierre del proyecto.

En el capítulo 1 se explica la importancia de implementar el Modelo de Certificación para el proceso de Pruebas No Funcionales de aquellos proyectos que contemplen la implementación de aplicaciones Web y Móviles desarrollada por GMD. Además, se plantea la problemática, los objetivos (tanto generales como específicos), indicadores que avalan el éxito de la implementación y los riesgos a los que se encuentran expuesto el proyecto.

En el capítulo 2, se presenta el cumplimiento de cada uno de los student outcomes de la carrera de Ingeniería de Sistemas de Información mediante el desarrollo de entregables del proyecto Modelo de Certificación de pruebas no funcionales.

En el capítulo 3 se define los principales conceptos como aplicaciones web y móviles, certificación, pruebas no funcionales, técnicas de diseño de pruebas y requerimientos de software. Teniendo definidos los conceptos se logra obtener un mejor entendimiento del proyecto.

En el capítulo 4, se muestra la investigación realizada de diversos casos de estudios relacionados a la certificación y buenas prácticas de las pruebas no funcionales, tomando como referencia lo desarrollado por autores correspondientes.

En el capítulo 5, se expone la elaboración del proyecto, desde la definición de los objetivos generales y específicos, hasta la elaboración del Modelo de Certificación y principales entregables que lo soportan.

El capítulo 6 contiene los resultados obtenidos producto de la implementación del modelo de pruebas no funcionales en la empresa GMD, mostrando los cuadros comparativos entre los resultados obtenidos antes y después de la implementación.

Finalmente, en el capítulo 7 define cómo se desarrolló el proyecto basado en el cumplimiento del cronograma, definición de principales recursos y los medios de comunicación definidos para entregar el producto final.

CAPITULO 1: DESCRIPCIÓN DEL PROYECTO

El presente capítulo está compuesto de 6 puntos que deben ser tomados en cuenta con el objetivo de tener un panorama general del proyecto. Primero, se describirá la problemática identificada en la empresa GMD. Posteriormente, se mencionarán cuál es el objetivo general del proyecto, objetivos específicos, indicadores de éxito y los beneficios que conlleva la implementación del modelo de certificación en la GMD. Además, se brindará detalle de la planificación del proyecto siguiendo los estándares y lineamientos del PMBOK.

Objeto de Estudio

El presente proyecto tiene como objetivo la implementación de un modelo de certificación para pruebas no funcionales en la empresa GMD, debido a que la empresa GMD solo se rige bajo un conjunto de actividades que no aseguran la calidad de sus pruebas. Por ello, el proyecto busca la implementación de un Modelo de Certificación para el proceso de validación y verificación dinámica en la etapa de Pruebas No Funcionales de los proyectos Web y móviles basándose en estándares, metodologías y buenas prácticas internacionales.

Dominio del Problema

GMD se ve en la necesidad de implementar una metodología de Pruebas no Funcionales para aplicaciones Web y Móviles, ya que ejecutan pruebas de requerimientos no Funcionales a través de un conjunto de actividades que no aseguran la calidad de despliegue de las aplicaciones.

Planteamiento de la Solución

La propuesta de solución que se plantea en el proyecto es la implementación de un modelo de certificación que ayudará a la empresa GMD a realizar las pruebas No funcionales de los proyectos que tengan en su alcance la implementación de una aplicación web y/o móvil basándose en metodologías, estándares y buenas prácticas internacionales de validación y verificación dinámica. Para efectos del desarrollo, se investigará en diversas fábricas de testing los modelos de pruebas a los que se rigen, y asimismo identificar que metodologías, estándares y buenas prácticas usan para que ayuden a construir el modelo a implementar. Este modelo a implementar será de gran utilidad debido a que se tendrá un modelo de certificación para las aplicaciones web y móviles en el proceso de Pruebas No Funcionales para futuras implementaciones.

Objetivos del Proyecto

Objetivo General

Implementar un Modelo de Certificación para el proceso de validación y verificación dinámica en la etapa de Pruebas No Funcionales de los proyectos de GMD basándose en estándares, metodologías y buenas prácticas internacionales.

Objetivos Específicos

Para poder cumplir con el objetivo general del proyecto, se plantean cinco objetivos específicos que serán detallados a continuación:

- OE1: Analizar el proceso de certificación de fábricas de testing con certificaciones internacionales, para identificar las actividades que se realizan e identificar las oportunidades de mejora.
- OE2: Diseñar un modelo de certificación basado en técnicas de validación y verificación dinámica para comprobar la calidad de las pruebas no funcionales sobre la empresa GMD.
- OE3: Validar el procedimiento de certificación de validación y verificación dinámica a implementar con proyectos que posea entre su alcance la implementación de aplicaciones web y/o móviles.
- OE4: Proponer un plan de continuidad del modelo de certificación basado en las técnicas de validación y verificación estática y dinámica para la empresa virtual IT-Services.

Indicadores de Éxito

El cumplimiento de los objetivos del proyecto será medido con los siguientes indicadores de éxito:

- I1. (OE1) Documento con el análisis FODA de los diferentes procesos de certificación de las fábricas de testing aprobado por el Cliente del Proyecto.
- I2. (OE2) Documento con el diseño y caracterización del modelo de certificación basado en las metodologías y buenas prácticas internacionales encontradas aprobado por el Cliente del Proyecto.

- I3. (OE2) Documento con indicadores definidos que aprueban la efectividad del modelo de certificación, aprobado por el Cliente del Proyecto.
- I4. (OE4) Documento que avale las pruebas realizadas a 2 requerimientos de la empresa GMD usando el modelo propuesto aprobado por el Cliente del Proyecto.
- I5. (OE3) Documento con la propuesta del modelo de certificación basado en la metodología y buenas prácticas en validación y verificación dinámica por parte de la empresa virtual IT-Services aprobado por el Profesor Cliente del Proyecto.

Planificación del Proyecto

Alcance

El alcance del proyecto incluirá:

- Identificar el estado actual de la empresa GMD sobre las pruebas No Funcionales de las aplicaciones Web y Móviles de tal manera que se propongan oportunidades de mejora.
- Realizar el análisis FODA de los procesos de certificación de las fábricas de testing.
- Implementación de un Modelo de Certificación de empresa GMD basándose en estándares, metodologías y buenas prácticas internacionales sobre las pruebas de validación y verificación dinámicas de los requerimientos no funcionales.
- Elaborar un listado con los indicadores para comprobar la calidad del Modelo de Certificación sobre las aplicaciones web y móviles de los diversos proyectos.
- Validar el Modelo de Certificación con proyectos que tengan en su alcance la implementación de aplicaciones web y/o móviles para comprobar calidad de la implementación.
- Implantar una herramienta para la gestión de incidentes basadas en las Pruebas No Funcionales para realizar el seguimiento de los errores encontrados durante la ejecución de las pruebas para requerimientos no funcionales.
- Proponer un Modelo de Certificación de técnicas de validación y verificación dinámica para la empresa virtual IT-Services.

Plan de Gestión del Tiempo

El Plan de Gestión del Tiempo del proyecto tiene como objetivo principal ayudar a gestionar los tiempos de los entregables que se define en el cronograma.

Fases del Proyecto

- La primera fase, inicio del proyecto, está compuesta básicamente por el siguiente entregable:
 - Project Charter
- La segunda fase, planificación del proyecto, cuenta con los entregables propios para la gestión del proyecto, tales como:
 - Cronograma del Proyecto
 - Descripción de Roles y Responsabilidades
 - Diccionario EDT
 - Matriz de Riesgos
 - Matriz RAM
 - Plan de Gestión de Calidad
 - Plan de Gestión de Comunicaciones
 - Plan de Gestión de Riesgos
 - Plan de Gestión de RRHH
 - Plan de Gestión del Cronograma
 - Plan de Gestión de Alcance
 - Registro de Interesados
- La tercera fase, ejecución del proyecto, se encuentra compuesta por los entregables propias de la solución del proyecto, tales como:
 - Documento de benchmarking

- Diseño y Caracterización del Modelo de Certificación
- Análisis FODA de Fábricas de Testing
- Especificación de Indicadores
- La cuarta fase, cierre del proyecto, se encuentra compuesto por los entregables correspondientes a la culminación del proyecto.
 - Acta de cierre y aceptación
 - Lecciones aprendidas
 - Memoria
 - Poster de proyecto

Plan de Gestión de Recursos Humanos

Ilustración 1: Organigrama de empresas virtuales de UPC

- **Comité de Dirección:** Encargados de aprobar los proyectos profesionales, así como del cumplimiento de los artefactos y entregables y de la decisión de la continuidad de los proyectos.
- **Cliente:** Monitorea los avances y brinda lineamientos durante todo el ciclo de vida del proyecto.
- **Gerente Profesor – DataCenter:** Monitorea los avances y brinda recomendaciones durante todo su ciclo de vida a los proyectos pertenecen a la cartera de la empresa.
- **Gerente General de Empresas Virtuales:** Conocer el estado de los proyectos de la cartera actual, las actividades de la Gerencia de Proyectos y Recursos, las nuevas propuestas de mejora y nueva cartera. Monitoreo y control de actividades programadas en el proyecto.

- Jefe del Proyecto: Planifica, dirige el proyecto y es el responsable del cumplimiento de los objetivos del mismo. Búsqueda, análisis y ejecución de la investigación de temáticas del proyecto.

Plan de Comunicaciones

El documento plan de gestión de comunicaciones tiene como objetivo definir los lineamientos para la comunicación entre el equipo del proyecto de DataCenter, el Cliente y todos los involucrados.

Guías para Reuniones

Todas las reuniones deben seguir las siguientes pautas:

- En caso de tener un nuevo miembro en cada reunión, se hará la respectiva presentación de este ante los asistentes.
- Se manejará una agenda para cada reunión de acuerdo a los temas de interés a tratarse, así como también las tareas pendientes de reuniones anteriores. Para este caso, se debe tener en cuenta el entregable a realizar según el diccionario EDT.
- Cada punto de la agenda contará con un nivel de prioridad a ser tratado.
- Se cuenta con un horario específico para cada reunión, sin embargo, será necesario confirmar/recordar la reunión vía correo electrónico un día antes como máximo.
- En caso se deba postergar una reunión, se deberá justificar y además acordar el nuevo día de esta.
- Al finalizar cada reunión se deberá recopilar lo tratado, resaltando los acuerdos para posteriormente elaborar el acta de reunión.
- Todas las actas de reunión deberán ser firmadas por el jefe de proyecto, asistente de proyecto y gerente general o cliente, respectivamente.

Guías para correo electrónico

Todas las remisiones de correos electrónicos deben seguir las siguientes pautas:

- Cada correo electrónico debe ser redactado de una manera formal.
- Todo correo electrónico al cliente debe ser copiado a todo el equipo de proyecto; en especial a gerente, jefe, asistente de proyecto

- Los correos electrónicos entre el gerente de la empresa y el cliente, deberá ser enviado por el jefe de proyecto para establecer un estándar en la conexión.
- En caso de envíos de solicitud de servicios a empresas externas, se debe copiar al correo de la empresa.

Plan de Gestión de Riesgos

Este plan permite identificar hechos que afecten el proyecto en sus tres aspectos: tiempo, alcance o recursos. Se encontraron 7 riesgos potenciales para el proyecto.

Tabla 1: Tabla de riesgos

#	Riesgo	Probabilidad	Impacto	Estrategia de mitigación
1	Cancelación de las reuniones con el cliente o profesor gerente	Bajo	Alto	Se pactarán reuniones por internet o durante la semana para subsanar este riesgo.
2	Cambio del alcance del proyecto por necesidades del cliente	Alto	Alto	Se realizará un ajuste al proyecto, considerando siempre el impacto en tiempo que este puede tener en los demás temas considerados dentro del alcance.
3	Incumplimiento de los entregables por parte del equipo	Medio	Alto	Se dedicará tiempos extras para cerrar los documentos pendientes.
4	Costo alto de la licencia de las herramientas a documentar	Medio	Medio	Se buscarán el financiamiento o préstamo.

#	Riesgo	Probabilidad	Impacto	Estrategia de mitigación
5	Disconformidad en la entrega de los informes ya desarrollados	Bajo	Alto	Se elaborará nuevamente el artefacto desarrollado, considerando las observaciones del Stakeholder involucrado.
6	La estimación del tiempo del proyecto es optimista, en vez de ser realista	Medio	Alto	Se debe elaborar el cronograma con tiempos realistas.
7	Incumplimiento de la corrección de los documentos a cargo del recurso de QS.	Medio	Medio	Se debe realizar el seguimiento de la solicitud de corrección enviada a través de correo electrónico.

CAPITULO 2: LOGROS DE LOS STUDENT OUTCOMES

En este capítulo se describe y detalla el logro de cada uno de los students outcomes estipulados por ABET. El objetivo principal de este programa establecido por ABET es evidenciar las habilidades, conocimientos y comportamientos que se adquieren en el desarrollo del proyecto Modelo de Certificación de Pruebas No Funcionales.

Student Outcome A

Aplica conocimientos de matemáticas, ciencias, computación e ingeniería.

Para el proyecto se aplica, en cuanto a conocimientos matemáticos refiere, el cálculo que se realiza sobre los indicadores definidos de manera que se determinará el score que ayuden a evidenciar que el modelo cumple con los objetivos planteados, esto se define en el documento de **Especificación de Indicadores**. Asimismo, se realizó el Benchmarking de Herramientas para las pruebas no funcionales de aplicaciones web y móviles, donde se analizó los alcances de cada una de manera que se logre seleccionar aquella que apoye en los resultados esperados que plantea el modelo de certificación. En lo que respecta a los principios de computación e ingeniería, hemos hecho uso de la norma ISO 29119, el cual brinda el Proceso Estándar de Calidad, del cual nos basamos para realizar el diseño del modelo de certificación, así mismo, nos apoyamos de la IEEE 829, el cual nos brinda la documentación para el proceso de pruebas de calidad, de manera que lo acoplamos al modelo y obtuvimos las plantillas que se usan durante la certificación. De la misma manera, tuvimos el apoyo de ISTQB, quien nos brindó los conceptos primordiales en cuanto a lo que a pruebas se refiere, de esta manera, logramos adaptarnos al contexto de calidad y logramos diseñar e implementar el modelo de Certificación para el proceso de Pruebas no funcionales. Esto se refleja en la sección **5.1.3 Proceso de Certificación de Pruebas No Funcionales** en la Memoria Final del Proyecto.

Student Outcome B

Diseña y conduce experimentos en base al análisis e interpretación de datos.

Para el presente proyecto y el cumplimiento de este Student Outcome, en base al análisis realizado de la situación actual en la que se encuentra la empresa GMD sobre el proceso que asegure la calidad del despliegue de los productos tanto web como móviles, se determinó que existe deficiencias por mejorar, y esto se detalla en el documento de **Especificación de Indicadores**. En este documento se realiza la comparación entre ambos frentes del proceso de pruebas, desde su modelo inicial de pruebas del que se regía el área de testing, hasta luego de la implementación del Modelo de Certificación, el cual se encuentra detallado en el documento **Diseño y Caracterización del Modelo de Certificación**, logrando obtener los resultados esperados cubriendo todos los aspectos relevantes en lo que respecta a pruebas no

Funcionales. De esta manera se puede constatar que la implementación se realizó de manera íntegra logrando cumplir con los objetivos del proyecto.

Student Outcome C

Diseña sistemas, componentes o procesos para encontrar soluciones en la atención de necesidades teniendo en cuenta restricciones económicas, sociales, políticas, éticas, de salud y seguridad y otras propias del entorno empresarial.

El proyecto considera la implementación de un modelo de Certificación de Pruebas No Funcionales para aquellos proyectos que en su alcance consideren la implementación de aplicaciones Web y/o Móviles, como solución a la brecha existente en el proceso de Pruebas de la empresa GMD.

Luego de realizar el análisis de la situación actual de la organización donde nos enfocamos principalmente en su proceso de pruebas no funcionales, determinamos que el modelo a implementar tiene como fin establecer un marco de trabajo enfocado principalmente en aquellos requerimientos no funcionales, donde se obtuvo que no se les da la consideración esperada, ya que solo se enfocan en la funcionalidad de la aplicación, mas no consideran los componentes que lo soportan. Estas consideraciones fueron plasmadas en los procesos diseñados, políticas establecidas y marco de trabajo documentado para su implementación de manera que se logre contemplar el marco de trabajo esperado. Lo expuesto anteriormente se encuentra detallado en el **Capítulo 5** de la presente memoria.

Ilustración 2: Representación gráfica del Student Outcome C

Student Outcome D

Participa en equipos multidisciplinarios desarrollando sus tareas eficientemente con profesionales de diferentes especialidades o dominios de aplicación.

Durante la fase de Diseño e Implementación del Modelo, se tuvo una activa participación de parte del Líder Técnico del Proyecto y jefe de infraestructura de GMD, quienes se propusieron a ayudar en la implementación de nuestro proyecto, ayudándonos en la mejora de los procesos diseñados basados en la experiencia que tienen. Esto se refleja en las **actas de aceptación de las pruebas** generadas luego de las reuniones para verificar las pruebas realizadas de GMD. Además de los colaboradores de GMD, también se tuvo interacción con los distintos Stakeholders del proyecto, como: Profesor Cliente, Profesor Gerente de DataCenter, Recurso de IT Service y del Revisor del Paper, quienes han tenido una participación activa durante la elaboración de los entregables del proyecto, y esto se refleja en las **actas de reunión** generadas durante el proyecto.

Ilustración 3: Representación gráfica del Student Outcome D

Student Outcome E

Identifica, formula y resuelve problemas de ingeniería.

El proyecto se sustenta a partir de la problemática identificada, como se puede apreciar en la ilustración 03, se observó que a partir de un conjunto de actividades que el área de testing de GMD ejecuta sobre el proceso de pruebas no funcionales, de ninguna forma se logra asegurar de ninguna forma la calidad de las pruebas ejecutadas. Como solución a esta problemática, se plantea el diseño e implementación de un **Modelo de Certificación** que se enfoque principalmente en el proceso de Pruebas No Funcionales de las aplicaciones Web y Móviles, el cual proporcione un esquema metódico de actividades, como se detalla en el documento **Diseño y Caracterización del Modelo de Certificación**. Además del esquema metódico, se logra realizar el análisis de resultados a partir de los proyectos proporcionados por la organización de manera que se logra identificar la brecha existente sobre el proceso de pruebas.

Ilustración 4: Representación gráfica del Student Outcome E

Student Outcome F

Propone soluciones a problemas de ingeniería con responsabilidad profesional y ética.

El proyecto desarrollado se plantea con el principal objetivo de cubrir una brecha existente en GMD frente a su proceso de pruebas no funcionales, pero para llevar a cabo el proyecto, nos exigían cumplir con las políticas organizacionales basadas en el **Código de Ética de Ingeniería del Software**, que menciona como principal ideología que los ingenieros de software deben verse obligados a hacer del análisis, especificación, diseño, desarrollo, pruebas y mantenimiento del software una profesión respetada y beneficiosa. Es por ello que nos recalcaron que el proyecto debe desarrollarse bajo los ocho principios, los cuales se detallan a continuación:

1. **Sociedad:** Los ingenieros de software actuarán de manera coherente con el interés social.
2. **Cliente y Empresario:** Los ingenieros de software actuarán de manera que produzca el mejor resultado para cliente y empresario, y de manera coherente con el interés social.
3. **Producto:** Los ingenieros de software garantizarán que sus productos y las modificaciones correspondientes cumplen los mayores estándares profesionales posibles.
4. **Valoración:** Los ingenieros de software mantendrán la integridad e independencia en sus valoraciones profesionales.
5. **Gestión:** Los líderes y gestores de ingeniería de software suscribirán y promoverán un enfoque ético en la gestión del desarrollo y mantenimiento del software.
6. **Profesión:** Los ingenieros de software avanzarán en la integridad y reputación de la profesión, de manera consistente con el interés social.
7. **Compañeros:** Los ingenieros del software serán justos y apoyarán a sus compañeros.

8. **Personal:** Los ingenieros del software participarán en el aprendizaje continuo referente a la práctica de su profesión y promoverán un enfoque ético en la práctica de la profesión.¹

Student Outcome G

Comunica ideas o resultados de manera oral o escrita con claridad y efectividad.

El presente Student Outcome se evidencia en la interacción periódica con los diferentes grupos de stakeholders, de los cuales, según nuestro enfoque, lo agrupamos en diferentes escenarios donde nos exponían su percepción sobre los avances correspondientes. El detalle a continuación:

- **Comité de Proyectos:** Encargados de evaluar la sustentación del desarrollo del proyecto y de emitir observaciones frente al cumplimiento de los objetivos planteados en las presentaciones definidas.
- **Calidad del Proyecto:** Se efectuaron coordinaciones de manera presencial y también mediante correos electrónicos con la empresa virtual IT-Service para la solicitud de recursos, de manera que se logre conseguir el aseguramiento de la calidad de los artefactos y pruebas que demandaba el desarrollo del proyecto.
- **Gestión del Proyecto:** Se efectuaron reuniones periódicas con el profesor Cliente y Gerente a través de mensajes instantáneas y videollamadas, y de manera presencial, respectivamente. Estas reuniones tenían como objetivos facilitarnos los principales lineamientos y alcance de nuestro proyecto, además de revisar y verificar los avances periódicos avalados por su juicio experto.
- **Stakeholders Externos (Empresa GMD – Integrantes del equipo de Testing):** En las reuniones acordadas, se logró conseguir la retroalimentación respectiva a las pruebas realizadas de su cartera de proyectos asignados. Asimismo, se logró definir el proceso de pruebas final gracias al conocimiento que ellos tenían al respecto.
- **Asesoría del Paper:** En las reuniones semanales efectuadas, se validaban los artículos científicos presentados, de manera que nos emita sus observaciones sobre las búsquedas realizadas y feedback, con el fin de que el Paper elaborado se encuentre alineado con proyecto.

¹ Código de Ética y Práctica de Ingeniería del Software de la ACM / IEEE Computer Society (1992)

Se evidencia el cumplimiento del student outcome a través de las **actas de reunión** sostenidas con los grupos definidos para el desarrollo del proyecto.

Ilustración 5: Representación gráfica del Student Outcome G

Student Outcome H

Identifica el impacto de las soluciones de Ingeniería de sistemas de información en el contexto global, económico y del entorno de la sociedad.

Tras la implementación de la solución propuesta, se concluye que el impacto generado en las organizaciones de desarrollo y testing es positivo, ya que durante su desarrollo se contempló el **uso de metodologías, estándares y buenas prácticas internacionales**, lo cual hace que el Modelo de Certificación se convierta en un estándar global para la ejecución de pruebas, además de ser **personalizable en su alcance y magnitud**. Asimismo, la implementación del modelo permite que la ejecución de las pruebas sea más eficiente logrando la **reducción de costos** en gran magnitud, ya que no se estaría invirtiendo en la implementación de herramientas y recursos caros que no logren contemplar los resultados esperados. Finalmente, en cuanto a la gestión de las pruebas se refiere, se logrará establecer

un canal de comunicación sostenible que conllevará a la adopción de una nueva ideología en cuanto a la ejecución de pruebas. Esto se aprecia en el **Capítulo 5** de la presente memoria.

Student Outcome I

Reconoce la necesidad de mantener sus conocimientos actualizados.

Para el desarrollo del proyecto requeríamos de información relevante que nos ayude a tener un mejor entendimiento del contexto del proyecto, por lo que nos fue necesario realizar una búsqueda exhaustiva de artículos científicos, libros de Calidad y otras fuentes bibliográficas que nos permitan mantener nuestros conocimientos actualizados y alineados. Con esto, podríamos mencionar que parte de nuestra actualización estuvo presente el libro del International Software Testing Qualifications Board (ISTQB), el cual nos dio a entender los Tipos de pruebas que actualmente existen en toda organización que contemple el desarrollo y testing de productos Software. Acotando el alcance de nuestro proyecto, nos esclareció que los requerimientos no funcionales representan el “cómo funciona el sistema”, diferenciándolo del “qué hace el sistema”. Asimismo, nos dio a entender la importancia del ciclo de pruebas el cual asegura la calidad del producto tanto en lo funcional como en lo no funcional. Además de este, hacemos referencia también al libro generado por el Software Engineering Institute, del cual obtuvimos aquellos requerimientos no funcionales establecidos y reconocidos mundialmente como principales a considerar para cualquier ciclo de pruebas no funcionales, de los cuales seleccionamos aquellos que la organización nos solicitó dar énfasis. Con ello elaboramos el modelo de Certificación, detallado en el **Diseño y Análisis de Modelo de Certificación**.

Por otro lado, se realizó la búsqueda de artículos científicos que hablen sobre las diversas pruebas no funcionales realizadas en diversas instituciones internacionales, así como sus casos de estudio. También se consideró la búsqueda de modelos de calidad elaborados e implementados que permitirán colocarnos en el contexto de lo a que certificación refiere. Esto nos dio una clara idea del alcance que debía tener el desarrollo del proyecto y se plasma en el **Paper** elaborado del correspondiente a nuestro proyecto.

Student Outcome J

Analiza hechos del mundo contemporáneo identificando el impacto en el desempeño profesional de Ingeniería.

El proyecto se sostiene bajo la necesidad de que las organizaciones deben tener una mayor consideración en cuanto a pruebas no funcionales refiere para las aplicaciones web y móviles que soportan las operaciones del negocio, esto conlleva a que se analice su importancia haciendo referencia a lo ocurrido en el año 1987 en Estados Unidos, el denominado el “Lunes Negro”, debido a que el índice Dow Jones se desplomó casi en un 22.6% mostrando debilidades en el sistema financiero, se concluyó que los sistemas se combinaron con la caída de los precios, lo cual agravó el problema, ya que los servidores eran incapaces de procesar el número de transacciones que recibía causando colapso y saturación del mercado. Esto le costó a Wall Street 500 Millones de dólares. Además de ello, por mencionar a organizaciones que se dedican al proceso de testing y fabricación de software, se obtuvo que no todas cumplen con un estándar de pruebas, sino que mayormente por juicio experto se ejecutan. Esto se evidencia en el documento **Análisis FODA de Fábricas de Testing y Software**, donde se realiza el análisis correspondiente del proceso de pruebas de calidad para requerimientos no funcionales.

Student Outcome K

Utiliza técnicas, herramientas y metodologías necesarias para la práctica de la ingeniería.

Para el desarrollo del proyecto, se gestionó bajo la metodología PMBOK, la cual nos permitió alinear y desarrollar el proyecto basado en las mejores prácticas, teniendo como revisor de los mismos al profesor gerente de DataCenter quien verificaba la consistencia de los entregables.

En cuanto a la elaboración del modelo de certificación, se optó por hacer uso de metodologías, estándares y buenas prácticas organizacionales, de manera que nos permita construir un modelo adaptable y flexible frente a la evolución de la tecnología. Para ellos nos basamos en la ISO 29119, el cual es un acuerdo internacional de estándares de pruebas de software que puede ser adoptado durante cualquier ciclo de vida del desarrollo del software. Además, nos basamos en la IEEE 829, el cual nos brinda la documentación primordial que todo proceso de pruebas debe considerar, de esta manera, acoplamos esta documentación con lo que obtuvimos las plantillas que se hacen uso durante el proceso de certificación de pruebas no funcionales.

También tuvimos de referencia a la ISTQB, que es una organización de certificación de la calidad de software, de donde obtuvimos los principales términos y conceptos de pruebas

para poder alinear el modelo a los objetivos planteados del proyecto. Asimismo, del SEI (Software Engineering Institute), organización que se encarga de desarrollar modelos de evaluación y mejora en el desarrollo de software, nos ayudó a comprender las consideraciones para elaborar un modelo flexible y adaptable.

Y en cuanto al modelamiento de los procesos que soporta el modelo, se usó el Bizagi Process Modeler, herramienta que nos ayudó a diagramar, documentar y simular procesos bajo la notación de BPMN, de manera que logramos optimizar los procesos gracias al levantamiento de información que realizamos y con ello logramos un modelo eficiente, flexible y adaptable. Esto se ve reflejado en el **Capítulo 5** de la presente memoria.

Student Outcome J – CAC

Comprende y brinda soporte para el uso, entrega y gestión de sistemas de información dentro de un entorno de sistemas de información.

Para implementar el modelo de certificación se realizaron modificaciones en los procesos que la soportan, logrando la flexibilidad esperada adaptándose a cambios solicitados por los integrantes del equipo de testing. Una vez aprobado el cambio, se procedió con la sustentación de implementación del modelo de certificación desarrollado, con el objetivo de que los stakeholders externos logren captar la importancia de tener una metodología de pruebas que asegure la calidad en el despliegue de las aplicaciones web y móviles, cubriendo así la brecha existente en la organización. Se explicaron los resultados obtenidos quedando conformes con lo expuesto. Para ello, se plasmó la capacitación realizada en un **acta de reunión** firmada por los principales stakeholders del proyecto.

WASC Student Outcome – Comunicación Escrita

Comunicación escrita es la capacidad para construir mensajes con contenido relevante, argumentos sólidos y claramente conectados adecuados a diversos propósitos y audiencia.

Para lograr este outcome, se desarrolló la presente Memoria del Proyecto. Esta contiene 7 capítulos estructurados que detallan la implementación del proyecto en mención. En el Capítulo 1 se mencionan los objetivos del proyecto (general y específico), indicadores de éxito y beneficios que se adoptan con la implementación del proyecto. El capítulo 2 describe como logramos cumplir con los Outcomes de ABET y WASC. El capítulo 3 relata los conceptos básicos indispensables para la comprensión del proyecto, los cuales son extraídos

de fuentes confiables. Respecto al capítulo 4, se detalla el análisis de artículos frente a la problemática encontrada para nuestro proyecto. Comprende cual es la situación actual de nuestro proyecto y de la posición de los diversos autores elegidos, con lo que logramos esclarecer el objetivo general y fortalecimos la implementación. El capítulo 5 comprende el desarrollo del proyecto en su totalidad, desde la investigación y análisis frente a la problemática del proyecto encontrada, hasta su implementación. El capítulo 6 muestra el detalle de los resultados obtenidos producto de la implementación, abarca los indicadores de éxito del modelo de certificación. Y por último, el capítulo 7, donde se abarca toda la gestión del proyecto, tales como las actividades, hitos, entregables, fechas, entre otros.

WASC Student Outcome – Comunicación Oral

Capacidad para transmitir oralmente mensajes de manera eficaz dirigidos a diversas audiencias, usando diferentes herramientas que faciliten su comprensión y el logro del propósito.

Este outcome se cumple con la interacción en fechas determinadas con los diferentes Stakeholders del proyecto, tales como el Comité de Proyectos, Cliente de Proyecto, Profesor Gerente, Stakeholders externos (empresa donde implementamos el proyecto) y el Asesor de Paper. Aquí se expone el avance del proyecto en su totalidad, desde como subsanar observaciones encontradas, delimitación del alcance del proyecto, evaluación del cumplimiento de objetivos, y exposición de resultados obtenidos como parte de la implementación realizada. Esto se evidencia con las actas de reunión sostenidas con los stakeholders mencionados.

WASC Student Outcome – Ciudadanía

Capacidad para valorar la convivencia humana en sociedades plurales, reflexionando acerca de las dimensiones morales de las propias acciones y decisiones, asumiendo la responsabilidad por las consecuencias en el marco del respeto de los derechos y deberes ciudadanos.

El desarrollo del proyecto contempló diversas coordinaciones, tanto con los stakeholders internos (Profesor Gerente, Cliente del Proyecto, Comité de Proyectos, Asesor de Paper) como externos (Integrantes del equipo de Testing), donde se exponía la necesidad de satisfacer el objetivo principal del proyecto basado en los objetivos específicos e indicadores de éxito. Estos acuerdos establecidos con ambas partes fueron apoyados por todos los

involucrados en el proyecto sin llegar al punto de determinar la cancelación del proyecto, sino al contrario, ayudar a cubrir la brecha que requerían satisfacer. Estos acuerdos contemplaban acuerdos contractuales como la cláusula de confidencialidad de resultados obtenidos, la empresa donde realizamos la implementación determinó que los resultados son internos y estaba prohibida su divulgación.

WASC Student Outcome – Pensamiento Innovador

Capacidad para detectar necesidades y oportunidades para generar proyectos o propuestas innovadoras, viables y rentables. Planifica y toma decisiones eficientes orientadas al objetivo del proyecto

Este outcome es contemplado con el desarrollo del proyecto, nuestro Modelo de Certificación de Pruebas no Funcionales contempla estándares y buenas prácticas internacionales con lo que logramos satisfacer nuestro objetivo principal. Se estableció la problemática inicial basado en el análisis FODA realizado a diversas empresas que laboraban bajo el mismo rubro, esto fue punto de partida para el desarrollo, y se logró cubrir la brecha identificada como parte de su proceso de pruebas. Demostramos en base a nuestros indicadores de éxito la viabilidad del proyecto, determinando valores reales basados en la información que nos proveían. Este cumplimiento se ve evidenciado en el **Capítulo 5 y 6** de la presente memoria.

WASC Student Outcome – Pensamiento Crítico

Capacidad para explorar de manera exhaustiva problemas, ideas o eventos para formular conclusiones u opiniones sólidamente justificadas.

Para el cumplimiento de este outcome se tuvieron 2 vertientes las cuales son expuestas a continuación:

1. **Objetivo General del Proyecto:** Consideramos prudente establecer objetivos específicos que ayuden a cumplir de manera granular con el objetivo principal, avalados claro está por indicadores de éxito (estos son expuestos en el **capítulo 1** de la presente memoria). Con ello, logramos realizar entrevistas a Fábricas de Software y Testing para entender su situación actual y así compararlo con nuestro objetivo planteado, se obtuvo de diversas fuentes bibliográficas las mejores prácticas y estándares internacionales que permitan construir un Modelo sólido y confiable.
2. **Artículos Científicos:** Como parte del desarrollo del proyecto, contemplamos también la búsqueda exhaustiva de artículos científicos relacionados con nuestro objetivo principal, asesorados por nuestro Co-Autor de Paper, quien nos ayudó a delimitar nuestro

alcance. Con ello, encontramos situaciones que coincidían con lo requerido, por lo que nos serían de utilidad para la construcción del modelo. Esto se ve contemplado en el **capítulo 4**.

WASC Student Outcome – Razonamiento Cuantitativo

Capacidad para interpretar, representar, comunicar y utilizar información cuantitativa diversa en situaciones de contexto real. Incluye calcular, razonar, emitir juicios y tomar decisiones con base en esta información cuantitativa

Para el desarrollo del proyecto, logramos hacer uso de varias herramientas que nos permitían determinar cuantitativamente los indicadores de éxito. Esto se denota en la **Especificación de Indicadores**, donde se expone el Benchmarking realizado para el uso de herramientas, el Análisis FODA generado producto de las entrevistas realizadas a Fábricas de Testing y Software. Adicionalmente a ello, consultamos a la ISTQB, ISO 29119, IEEE, SEI, entre otros, quienes nos brindaron los conceptos primordiales en cuanto a lo que a pruebas se refiere, de esta manera, logramos adaptarnos al contexto de calidad y logramos diseñar e implementar el modelo de Certificación para el proceso de Pruebas no funcionales. Esta investigación se vio reflejada en la sección **5.1.3 Proceso de Certificación de Pruebas No Funcionales** donde se detalló el flujo final que cumplirá el Modelo de Certificación. Todo esto determinó la viabilidad económica del proyecto expuesto en el **capítulo 6**, donde se detalla los principales beneficios de la implementación.

WASC Student Outcome – Manejo de la Información

Capacidad de identificar la información necesaria, así como de buscarla, seleccionarla, evaluarla y usarla éticamente, con la finalidad de resolver un problema

Para el cumplimiento de este outcome se definió la situación inicial del proyecto y de la situación actual en la que se encontraba la empresa donde se realizaría la implementación, de esta manera se determinó la brecha existente. Antes de iniciar la implementación, la empresa nos propuso acatar los puntos de confidencialidad como parte de su política en cuanto a la divulgación de los resultados obtenidos, a lo que accedimos de manera íntegra. Durante el desarrollo del proyecto, la comunicación era fluida lo que facilitó el rápido entendimiento de la brecha existente. Resultado de la implementación, validamos los modelos basados en el Juicio de Expertos, con lo que obtuvimos resultados esperados, dejando evidenciado el cumplimiento de los indicadores de éxito, así como cubrir la brecha identificada. Esto se evidencia en el **capítulo 6** de la presente memoria.

CAPITULO 3: MARCO TEÓRICO

En este capítulo se define los conceptos básicos necesarios para la comprensión del proyecto.
Se detalla los conceptos relacionado al proceso de desarrollo de aplicaciones móviles y web.

Certificación

Definición según AENOR

La certificación es la acción llevada a cabo por una entidad independiente de las partes interesadas mediante la que se manifiesta que una organización, producto, proceso o servicio, cumple los requisitos definidos en unas normas o especificaciones técnicas.²

Definición según ISO

La certificación es un procedimiento mediante el cual un tercero otorga una garantía escrita de que un producto, elaboración o servicio está en conformidad con ciertas normas.

Definición según ISTQB

El proceso de confirmación de que un componente, sistema o persona cumple con los requisitos especificados.³

Síntesis

Basado en las definiciones anteriores, podemos deducir que el objetivo final de cada una no varía, por lo que podríamos concluir que Certificación se define como aquel proceso por el que se confirma que una empresa u organización, producto, proceso o persona cumple con los estándares y requisitos establecidos. Habiendo definido certificación en un contexto abierto, para el contexto de pruebas de software, podríamos definir como el proceso mediante el cual se confirma que un producto software cumple con los requisitos establecidos.

² Cfr.AENOR 65

³ Cfr.ISTQB

Tipos de Aplicaciones

Aplicaciones Web

Una aplicación web es una herramienta desarrollada en lenguajes como lo son HTML o JavaScripts, la cual es publicada en un servidor web. Esta se caracteriza por la facilidad que tiene el usuario a acceder a través de un navegador web mediante la conexión a internet.

Ventajas de las Aplicaciones Web

Las ventajas que se obtienen son:

- Las actualizaciones que se implementen sobre las aplicaciones web se instalan automáticamente, de tal manera que el usuario no se vea afectado por los cambios a realizar.
- El acceso a la aplicación es en cualquier momento y desde cualquier dispositivo que tenga el acceso a un navegador web.
- Facilidad de conexión de múltiples usuarios.
- No requiere de una instalación para poder operar en ella.

Desventajas de las Aplicaciones Web

- La compatibilidad se denota siempre y cuando el navegador se encuentre actualizado, independientemente del sistema operativo actual.
- Se necesita indispensablemente una conexión a Internet para poder realizar las operaciones en el aplicativo web.
- Se dependen de algunos plugins que se configuran en el mismo navegador.

Aplicaciones Móviles

Es aquella herramienta diseñada exclusivamente para los celulares inteligentes, tablets u otros tipos de dispositivos móviles, con la finalidad de ayuda al usuario a realizar una tarea en específica de cualquier índole (noticias, ocio, etc.). Estas herramientas son desarrolladas en los distintos lenguajes compatibles actualmente en el mercado, como lo son Android, Apple y Microsoft Windows Phone.

Aplicaciones Nativas

Son las aplicaciones desarrolladas específicamente para un sistema operativo determinado, ya sea del sistema operativo Android, IOS, etc.

Ventajas de las Aplicaciones Nativas

- Códigos de programación reutilizable.
- Desarrollo más sencillo y de menor costo
- Las aplicaciones se distribuyen a través de los stores propios de los fabricantes.

Desventajas de las aplicaciones Nativas

- Se necesitan desarrolladores especializados en esos lenguajes de programación.
- Las aplicaciones son creadas para un sistema operativo en específico.
- Necesita una aprobación de los fabricantes para que sean publicadas en los stores.

Aplicaciones Web Móviles

Son aplicaciones desarrolladas en HTML, JavaScript apoyado del CSS los cuales solo necesitan de un navegador para poder ser ejecutadas.

Ventajas de las aplicaciones Web Móviles

- Se puede tener acceso y usar desde cualquier dispositivo que posea navegador.
- Funciona sin algún sistema operativo en específico.
- El desarrollo de éstas es más sencillo

Desventajas de las aplicaciones Web Móviles

- No se ejecuta de la misma manera en que se ejecutaría desde una PC.
- Requieren de una conexión a internet
- No es amigable al usuario debido a la usabilidad.

Aplicaciones Híbridas

Son aplicaciones que combinan ambos tipos de aplicaciones explicados anteriormente, son desarrolladas para un sistema operativo en específico, y a la vez se encuentra desarrollada para el uso desde un navegador.

Ventajas de las aplicaciones Híbridas

- Ambos son de uso gratuito, solo se debe descargar.
- El costo del desarrollo es menor al que se tiene por desarrollar por separado.

Desventaja de las aplicaciones Híbridas

- Es muy complicado el desarrollo de este tipo de aplicaciones.
- La experiencia del usuario no es la esperada.

Pruebas no Funcionales

Clasificación según SWEBOK

Pruebas de Regresión

Es la repetición de pruebas selectivas de un sistema o componente para verificar que las modificaciones no han causado efectos no deseados y que el sistema o componente sigue cumpliendo con sus requisitos especificados.

Pruebas de Performance

Las pruebas de rendimiento verifican que el software cumple con los requisitos de funcionamiento específicos y evalúa características de rendimiento- por ejemplo, la capacidad y el tiempo de respuesta.

Pruebas de Seguridad

Las pruebas de seguridad se centran en la verificación de que el software está protegido de ataques externos. En particular, las pruebas de seguridad verifican la confidencialidad, integridad y disponibilidad de los sistemas y sus datos. Por lo general, las pruebas de seguridad incluyen la verificación contra el mal uso y el abuso del software o sistema (pruebas negativas).

Pruebas de Estrés

Las pruebas de estrés ejercen de software en la carga máxima de diseño, así como más allá de ella, con el objetivo de determinar los límites de comportamiento, y para poner a prueba los mecanismos de defensa en sistemas críticos.

Pruebas de Configuración

En los casos en que el software se construye para servir a diferentes usuarios, pruebas de configuración verifica el software bajo diferentes configuraciones especificadas.⁴

Clasificación según ISQTB

Pruebas de Aceptación

Son las pruebas formales con respecto a las necesidades del usuario, requisitos y procesos de negocio, las cuales son llevadas a cabo para determinar si es o no un sistema que satisface los criterios de aceptación y la habilitación por el usuario, los clientes u otra entidad autorizada para determinar si acepta o no el sistema.

Pruebas de Carga

Es un tipo de pruebas de rendimiento que se lleva a cabo para evaluar el comportamiento de un componente o sistema con el aumento de la carga, por ejemplo, número de usuarios y / o el número de transacciones paralelas, para determinar lo que la carga puede manejar por el componente o sistema.

Pruebas de Regresión

Son las pruebas de un programa previamente probado siguiendo las modificaciones para asegurar que los defectos no se han introducido o descubierta en zonas no modificadas del software, como resultado de los cambios realizados. Se realiza cuando se cambia el software o de su entorno.

Pruebas de Estrés

Son las pruebas de rendimiento realizado para evaluar un sistema o componente en o más allá de los límites de sus cargas de trabajo anticipadas o especificadas, o con menor disponibilidad de recursos tales como el acceso a la memoria o servidores.

⁴ SWEBOK cap 5 : 77

Pruebas de Usabilidad

Son las pruebas para determinar el grado en que se entiende el producto de software, fácil de aprender, fácil de operar y atractivo para los usuarios en condiciones especificadas.

Pruebas de Mantenimiento

Son las pruebas de los cambios en un sistema operativo o el impacto de un cambio de entorno a un sistema operativo.

Pruebas de Fiabilidad

Estas pruebas sirven para determinar la fiabilidad de un producto de software. Prueba la capacidad del producto de software para llevar a cabo sus funciones requeridas bajo las condiciones establecidas por un período de tiempo determinado, o para un número determinado de operaciones.

Pruebas de Características del Producto Software (No funcionales)

Las pruebas no funcionales tienen como objetivo medir las características del sistema, las cuales pueden cuantificarse para ser comparadas con una escala variable definida. Es decir, se orientan al “cómo funciona el sistema”.

Dentro de este tipo se incluye una serie de pruebas, dentro de las cuales las más usadas son:

- Rendimiento
- Carga
- Estrés
- Usabilidad
- Mantenibilidad
- Confiabilidad
- Portabilidad ⁵

⁵ Cfr. Graham Evans, Black 2011

Pruebas de la Estructura o Arquitectura del Software

Las pruebas de estructura o arquitectura del software tienen como objetivo medir el grado en el que un conjunto de casos de prueba ha llegado a cubrir a una estructura evaluada.

De esta manera, se puede medir el rigor de los casos de prueba, para que, en caso no logré una gran cobertura dentro de la estructura, se defina otros que permitan cubrirla

Pruebas Relacionadas a los Cambios (Confirmación y Regresión)

- Pruebas de confirmación

Las pruebas de confirmación tienen como objetivo confirmar que los defectos reportados han sido corregidos en su totalidad. Cabe hacer énfasis que esta prueba solo se centra que el defecto encontrado ya no se muestre.

- Pruebas de Regresión

Las pruebas de regresión tienen como objetivo verificar que los cambios que se hayan producido en un sistema, software, o en el entorno no causaron efectos secundarios que afecten el funcionamiento esperado.

Para la realización de estas pruebas se consideran a las mencionadas en los puntos anteriores: Funcionales, No funcionales y Estructurales.

Clasificación según BURNSTEIN

Pruebas Funcionales

Las pruebas funcionales tienen como objetivo principal asegurar que el funcionamiento del sistema se encuentre acorde con los requisitos especificados.

Pruebas de Rendimiento

Las pruebas de rendimiento tienen como objetivo demostrar los niveles de calidad en cuanto al cumplimiento de los requisitos de rendimientos que se requieran, los cuales deben ser cuantitativos. Algunos de los aspectos a medir son el tiempo de respuesta de una acción, uso de memoria, through-out, retrasos, entre otros.

Pruebas de Estrés

Las pruebas de estrés tienen como objetivo “romper el sistema”, es decir, hallar aquellas circunstancias en las que el sistema es susceptible y ya no responda. Esto se logrará en base

a la carga que reciban, tomando en cuenta los niveles de carga definidos. Entre los aspectos que se toman en cuenta se encuentra las condiciones de ejecución, los tiempos muertos, agotamiento de recursos, entre otros.

Pruebas de Configuración

Las pruebas de configuración tienen como objetivo evaluar la disponibilidad y rendimiento cuando se haya realizado un intercambio de hardware o reconfiguración.

Pruebas de Seguridad

Las pruebas de seguridad tienen como objetivo evaluar las características del sistema, dentro de las cuales se encuentran aspectos como la disponibilidad, integridad y confidencialidad de los datos y servicios.

Pruebas de Recuperación

Las pruebas de recuperación tienen como objetivo evaluar que un sistema es capaz de recuperarse luego de la pérdida de recursos. Las pruebas permitirán establecer si el sistema retorna a funcionar de manera estable y que ninguna de las transacciones que se encontraban realizando se hallan visto afectadas.

Tipos de Pruebas por Ejecución

Pruebas Manuales

Este tipo de pruebas son ejecutados por un tester quien, sin la ayuda de alguna herramienta especializada u automatizada, sigue una serie de pasos definidos en un plan de pruebas. Este test plan indica la forma en que se debe de desarrollar las pruebas de tal manera que se pueda comprobar el resultado esperado.

Pruebas Automatizadas

Definición según ISTQB

La automatización de pruebas ayuda a ejecutar muchos casos de pruebas consistente y repetidamente en diferentes versiones del SUT (SystemUnder test). Pero la automatización

de pruebas es más que un mecanismo para la ejecución de un conjunto de pruebas y que, en muchos casos, no necesita la interacción humana.⁶

Definición según SWEBOK

Las pruebas automatizadas apoyan a la ejecución de las definiciones de proceso o proporcionan orientación a los humanos a realizar dichos procesos. Además, hay herramientas que pueden ejecutar procesos definidos para brindar un soporte automatizado.⁷

Objetivo de las pruebas automatizadas

El objetivo de las pruebas automatizadas es la de simplificar la mayor parte del esfuerzo del tester con ayuda de herramientas automatizadas y especializadas para cada tipo de aplicación o sistema. Asimismo, las herramientas pueden ayudar a ejecutarse las pruebas repetidamente a cualquier hora del día.

Problemas con la automatización de pruebas

Según Bret Pettichorde numera algunos problemas que aquejan a la automatización. Siguen siendo aplicables, pero están destinados a equipos o aplicaciones que no incorporan la automatización como parte de su desarrollo. Existe una lista de Bret para mostrar lo que es probable que tenga problemas si no se incluye la automatización como parte de los entregables de los proyectos cotidianos.

- Falta de objetivos claros
- Falta de experiencia
- Existe alta rotación, ya que se pierde la experiencia que se puede tener.
- Centrándose en la solución del problema la tecnología puede hacer que usted pierda de vista si el resultado responde a la necesidad de pruebas.

⁶ Cfr. ISTQB Glosario

⁷ Cfr. SWEBOK

Técnicas de Diseño de Pruebas

Clasificación según ISTQB⁸

Pruebas de Caja Blanca

Es un método de diseño de casos de prueba que usa la estructura de control del diseño procedimental para derivar los casos de pruebas.

Este tipo de pruebas garantiza:

- Se ejecutan al menos una vez todos los caminos independientes de cada módulo
- Se utilizan las decisiones en su parte verdadera y en su parte falsa
- Se ejecuten todos los bucles en sus límites
- Se utilizan todas las estructuras de datos internas

Prueba del Camino Básico

Permite obtener una medida de la complejidad de un diseño procedimental, y utilizar esta medida como guía para la definición de una serie de caminos básicos de ejecución, diseñando casos de prueba que garanticen que cada camino se ejecuta al menos una vez.

Prueba de bucles

Es una técnica de prueba de caja blanca que se centra en la validez de las construcciones de los bucles.

Pruebas de Caja negra

Esta técnica se basa en la especificación del software, también, conocido como técnica de caja negra o técnicas de prueba de entrada / salida. En esencia, el probador se concentra en lo que hace el software más no como lo hace.⁹

Las pruebas de caja negra permiten demostrar que:

- Las funciones del software son operativas.

⁸ Cfr. ISTQB

⁹ Cfr. ISTQB 2008 : 86

- La entrada se acepta de forma correcta.
- Se produce una salida correcta.
- La integridad de la información externa se mantiene.

Prueba de Participación equivalente

Este método de prueba de caja negra divide el dominio de entrada de un programa en clases de datos, a partir de las cuales deriva los casos de prueba.

Técnica de prueba basada en experiencia

En las técnicas basadas en la experiencia, el conocimiento, las habilidades y los antecedentes de las personas son un principal contribuyente a las condiciones de prueba y casos de prueba. La experiencia de las dos personas técnicas y de negocios es importante, ya que aportan diferentes perspectivas en el análisis de la prueba y el proceso de diseño. Debido a la experiencia previa con sistemas similares, pueden tener una visión de lo que podría salir mal, lo que es muy útil para la prueba.

Técnica de Pruebas Estáticas

Las técnicas de pruebas estáticas no ejecutan el código examinado y son generalmente usadas antes de que cualquier prueba sea ejecutada en el software. ¹⁰La mayor parte de pruebas estáticas pueden ser utilizadas para "probar" cualquier forma de documento incluido el código fuente, documentos de diseño y modelos, especificaciones funcionales y las especificaciones de requisitos. La mayoría de las veces son usadas estáticamente para probar el código fuente.¹¹

¹⁰ Cfr : ISTQB 2008 : 86

¹¹ Cfr : ISTQB 2008 : 87

Requerimientos de Software

Clasificación según SWEBOK¹²

Requerimientos Funcionales

Son aquellos que describen la funcionalidad o características que el sistema ejecutará. También pueden ser considerados como un conjunto finito de pasos escritos que se utilizar para validar su comportamiento.

Requerimientos No Funcionales

Son aquellos que actúan para restringir la aplicación o como requisitos de calidad. Dentro de estos se encuentran considerados los siguientes:

- Rendimiento
- Mantenimiento
- Integridad
- Fiabilidad
- Seguridad
- Interoperabilidad, etc.

Clasificación según CMMI - DEV

Requerimientos del Cliente

Son aquellos requerimientos que resultan de la consolidación de las necesidades, expectativas, restricciones e interfaces establecidas por el cliente. ¹³

Requerimientos del Producto

Son los requerimientos del cliente analizados y llevados a un nivel más detallado y preciso. Es decir, estas son llevadas y asociadas con cada una de las fases del ciclo de vida del producto y bajo un lenguaje técnico comprensible para los desarrolladores.

¹² Cfr. SWEBOK

¹³ Cfr. CMMI

Requerimientos de Componentes del Producto

Son los requerimientos donde se brinda una especificación completa de un componente de producto, en este se incluye la forma, el ajuste, función, rendimiento, entre otros.

Clasificación según ISQTB

Requerimientos Funcionales

Son aquellos que especifican una función que un componente o sistema debe cumplir.

Requerimientos No Funcionales

Son aquellos que no se refieren a la funcionalidad, pero si a los siguientes atributos:

- Fiabilidad
- Eficiencia
- Usabilidad
- Mantenibilidad
- Portabilidad¹⁴

Modelos Referenciales

V-MODEL, ISTQB

V-Model o Model en V, significa Modelo de Verificación y Validación. Al igual que el modelo de cascada, el ciclo de vida en forma de V es una ruta secuencial de ejecución de procesos. Cada fase debe completarse antes de que comience la siguiente fase. Las pruebas del producto se planifican en paralelo con una fase correspondiente de desarrollo en el modelo V.¹⁵

En la práctica, un modelo V puede tener más, menos o diferentes niveles de desarrollo y pruebas, dependiendo del proyecto y del producto de software. Por ejemplo, puede haber pruebas de integración de componentes después de la prueba de componentes y pruebas de integración de sistemas después de las pruebas del sistema.

¹⁴ Cfr. ISTQB

¹⁵ Cfr. ISTQB

Los productos de trabajo de software (como escenarios empresariales o casos de uso, especificaciones de requisitos, documentos de diseño y código) producidos durante el desarrollo son a menudo la base de las pruebas en uno o más niveles de prueba. Las referencias para productos de trabajo genéricos incluyen CMMI (Capability Maturity Model Integration) o 'Procesos del ciclo de vida del software' (IEEE / IEC 12207). La verificación y validación (y el diseño de la prueba temprana) se pueden llevar a cabo durante el desarrollo de los productos de trabajo de software.¹⁶

Ilustración 6: V- Model

Ventaja del Modelo en V

- Simple y fácil de usar.
- Las actividades de prueba como la planificación, el diseño de pruebas ocurren mucho antes de la codificación. Esto ahorra mucho tiempo. De ahí una mayor probabilidad de éxito sobre el modelo de cascada.
- El seguimiento proactivo de defectos, es decir, los defectos, se encuentran en las primeras etapas.
- Evita el flujo descendente de los defectos.
- Funciona bien para pequeños proyectos donde los requisitos son fácilmente comprendidos.

¹⁶ Cfr. ISTQB

Desventajas del Modelo en V

- Muy rígido y menos flexible.
- El software se desarrolla durante la fase de implementación, por lo que no se producen prototipos tempranos del software.
- Si se producen cambios a mitad de camino, los documentos de prueba junto con los documentos de requisitos deben actualizarse.

En cualquier modelo de ciclo de vida, hay varias características de las buenas pruebas:

- Para cada actividad de desarrollo hay una actividad de prueba correspondiente.
- Cada nivel de prueba tiene objetivos de prueba específicos para ese nivel.
- El análisis y diseño de las pruebas para un nivel de prueba dado debe comenzar durante la actividad de desarrollo correspondiente.
- Los probadores deben participar en la revisión de documentos tan pronto como los borradores estén disponibles en el ciclo de vida del desarrollo.

Los niveles de prueba se pueden combinar o reorganizar dependiendo de la naturaleza del proyecto o de la arquitectura del sistema.

UML Testing Profile

El UML 2.0 Testing Profile proporciona conceptos que apuntan al desarrollo pragmático y sistemático de especificaciones de prueba concisas y modelos de prueba para las pruebas de caja negra y caja gris. En particular, el perfil introduce conceptos que abarcan: arquitectura de prueba, comportamiento de prueba, datos de prueba y tiempo. Juntos, estos conceptos definen un lenguaje de modelado para visualizar, especificar, analizar, construir y documentar los artefactos de un sistema de prueba.¹⁷

Existen varios enfoques para la integración de la información relacionada con las pruebas en los desarrollos del sistema con UML. De acuerdo con la metodología de prueba clásica, hay que distinguir dos aspectos diferentes: el modelado de las características relacionadas con el

¹⁷ <http://utp.omg.org/>

sistema (es decir del sistema bajo prueba, el SUT) y la definición de las características del modelo de prueba (es decir del sistema de prueba).

Un enfoque se denomina prueba de integración. Adopta la sintaxis UML y utiliza diagramas de diagrama de estado para los objetos bajo prueba. El objetivo de las pruebas de integración es minimizar los costos, tiempo y esfuerzo de las pruebas, es decir, desarrollar inicialmente controladores de prueba personalizados, talones de prueba y casos de prueba y adaptarlos y volver a ejecutarlos repetidamente para fines de pruebas de regresión en cada nivel de integración.¹⁸

Ilustración 7: UML Testing Profile

ISO/IEEE 29119 – SOFTWARE TESTING STANDARD

Es un conjunto internacionalmente acordado de estándares para pruebas de software que se pueden utilizar en cualquier ciclo de vida de desarrollo de software u organización. Mediante la implementación de estas normas, usted estará adoptando los únicos estándares

¹⁸ Cfr. UML 2.0

internacionalmente reconocidos y acordados para las pruebas de software, que proporcionarán a su organización un enfoque de alta calidad para las pruebas que se pueden comunicar en todo el mundo.¹⁹

Ilustración 8: Modelo de Referencia – ISO 29119

Este nuevo conjunto de estándares internacionales pretende ser lo más inclusivo posible, por lo que los estándares son genéricos y son capaces de soportar las pruebas en una amplia variedad de dominios de aplicación y para diferentes niveles de criticidad. Si las normas reglamentarias relacionadas con la seguridad exigen que realice pruebas para alcanzar un cierto nivel de cobertura, estas nuevas normas proporcionan procesos de prueba a seguir y definen las técnicas de diseño de la prueba y las medidas de cobertura que debe utilizar y lograr, respectivamente. Por el contrario, las normas también se han escrito para asegurar que los que utilizan pruebas exploratorias no están excluidos de cumplir con las normas si ellos (o sus clientes) así lo desean. Las normas no son específicas de un ciclo de vida en particular, por lo que son aplicables a los ensayos en proyectos que utilizan enfoques secuenciales, iterativos y ágiles

¹⁹ Cfr. ISO 29119 Software Testing Standard

CAPITULO 4: ESTADO DEL ARTE

En este capítulo se muestran diversos análisis de autores que aportaran al proyecto. Además, se da énfasis en las propuestas de estos, ya que en base a eso se pretende enriquecer el proyecto con un enfoque que aún no se ha visto.

Problema

Actualmente, las aplicaciones web y móviles son parte de la vida de las personas. Su desarrollo de una aplicación móvil implica el análisis, construcción y diseño de esta. Las pruebas, a su vez, se han convertido en una actividad de suma importancia dentro del ciclo de vida del software, ya que permite el aseguramiento de la calidad del producto. Por ello, las pruebas a aplicativos web y móviles requieren la aplicación de técnicas, herramientas y métodos de pruebas.

Importancia del problema

En muchos proyectos de desarrollo de software, las pruebas antes de lanzar al mercado un aplicativo nuevo no son tomadas en cuenta por la complejidad. Se ha detectado casos en los que el proyecto fracasó debido a que no se contempló la realización de pruebas en los requerimientos no funcionales. Uno de ellos es el sistema desarrollado para el departamento de Inteligencia del Ejército de Estados Unidos que presentó fallas en el factor de capacidad, rendimiento y usabilidad, dando como consecuencia una pérdida de hasta \$2.7 mil millones de dólares²⁰.

Un estudio realizado a usuarios de aplicaciones móviles revela que el 56% de estos usuarios han tenido problemas cuando han usado aplicaciones en un dispositivo móvil ya sea por ser congelado, salió un error inesperado, consumo excesivo de batería, entre otros errores que experimentaron²¹. Por este motivo, se realizará una investigación que tiene como objetivo conocer cómo reducir la complejidad y duración de la certificación de pruebas para que las aplicaciones móviles desarrolladas lleguen al usuario final con la menor margen de errores.

Revisión de literatura

Song, H. et al (2011) propone desarrollar un marco integrado de pruebas automatizadas para que los casos de prueba puedan ser probados en plataformas heterogéneas debido a la variedad de plataformas en aplicaciones móviles que requieren ingenieros de pruebas que inviertan tiempo y esfuerzo para testear aplicaciones en cada plataforma.

²⁰ Evaluating the Model-Based Testing Approach in the Context of Mobile Applications

²¹ <http://www.dynatrace.com/es/index.html>

Amalfitano, D. et al (2011) presentan la automatización de las pruebas para aplicaciones móviles a través de una técnica y una herramienta para llevar a cabo las pruebas crash, pruebas de regresión y la generación automática de casos de prueba.

Villanes, I.K. et al (2015) presenta un marco de trabajo que proporciona pruebas automatizadas para aplicaciones móviles y se basa en tecnología cloud. Emula dispositivos móviles sobre máquinas virtuales e infraestructura cloud. El estudio concluye con la validación del marco a través de experimentos, donde se observa que los dispositivos emulados podrían ser probados utilizando las pruebas de instalación.

Lu, L. et al (2012) realiza un estudio acerca de las aplicaciones móviles, web y tradicionales, en donde encuentran marcadas diferencias entre ellas por las características físicas de los dispositivos móviles. Por este motivo, plantea un método para la realización de pruebas automatizadas basadas en pruebas funcionales para generar casos de prueba con la ayuda de emuladores móviles.

Lettner, F. et al (2012) observa que existe una brecha para la evaluación de interfaces de aplicaciones móviles para la facilidad de uso. Por ello, presenta una serie de herramientas para la evaluación automatizada de interfaces que es mapea las interacciones del usuario en una aplicación móvil durante el ciclo de vida del producto software.

Nagowah, L. et al (2012) presenta un marco de trabajo para la automatización pruebas de conectividad y rendimiento en dispositivos móviles en lugar de realizarse en emuladores. En él se describe que pruebas automatizadas en emuladores puede poner en duda la fiabilidad de la prueba realizada, ya que no utiliza recursos propios de un dispositivo móvil.

Adamsen, C.Q. et al (2015) menciona que asegurar la calidad y la eficacia de los casos de prueba tiene un alto grado de complejidad. Por esta razón, propone un modelo de pruebas basado en casos de pruebas existentes. De esta manera, la aplicación es expuesta a eventos inesperados en el ambiente de pruebas.

Hay, R. et al (2015) brinda una metodología para la detección de vulnerabilidades de comunicación en aplicaciones móviles que funcionan en el sistema operativo para móviles Android. Adicionalmente, propone un listado de 8 tipos de vulnerabilidades concretas que puedan dar como resultado el erróneo manejo de los mensajes de comunicación entre las aplicaciones de dispositivos móviles.

Vieira, V. et al (2015) propone como aporte estrategias para simular escenarios de pruebas en dispositivos móviles como mensajes, llamadas, wifi, GPS, entre otros; según el contexto de cada aplicación (SUT) debido a que las aplicaciones móviles toman mayor acogida y ,para evitar fracasos una vez publicada en el mercado, tienen que pasar por una etapa de pruebas.

Nuo Li et al (2004) propone un marco de trabajo que tiene como finalidad orientar y guiar las actividades de testeo para poder ejecutar una satisfactoria prueba a una aplicación web. Este marco consiste en 4 fases, Pruebas de análisis de requerimientos, Generación y selección de casos de prueba, pruebas de ejecución y resultados de pruebas.

Derek Flood et al (2012) presenta un estudio en el que se examinaron aspectos de la interacción del usuario con las aplicaciones móviles durante las diversas etapas del ciclo de vida del uso de aplicaciones móviles: identificación, instalación, uso y eliminación de la aplicación. En base a ello, propone algunas directrices para la usabilidad de una aplicación móvil, tales como Ubicación App, Uso de App y Remoción App dentro ellas describen cuales son las características de cada una respectivamente.

Denni Aldi Ramadhani et al (2015) presenta un sistema de clasificación de requerimientos no funcionales que se dividen en tres fases principales. La automatización del etiquetado de los datos de entrenamiento, la medida de la relación sintética, y el proceso de clasificación que comprende las etapas de la formación y la clasificación de los requerimientos.

Motivación

Se han diseñado diversas técnicas, metodologías y marcos de trabajo enfocados en pruebas funcionales y no funcionales para la ejecución del proceso de testing y de esta manera, reducir las fallas para asegurar la calidad de las aplicaciones web y móviles. Sin embargo, no hay un enfoque donde se contemple todas las características y requerimientos no funcionales que se requieren cubrir tanto para los dispositivos como para las aplicaciones web y móviles.

Aporte

Basándose en la ISO 29119, se propone un modelo de certificación de requerimientos no funcionales que posee 4 capas y que se encuentra dentro de un ciclo de mejora de la calidad. Las actividades a realizar en cada capa están enfocadas en la gestión organización del

proceso de pruebas, la gestión del proceso de pruebas y ejecución de las pruebas de los requerimientos no funcionales.

CAPITULO 5: DESARROLLO DEL PROYECTO

En este capítulo se define los conceptos contemplados dentro del modelo de certificación diseñado para la ejecución de pruebas no Funcionales. Adicionalmente, se mostrará el modelo propuesto a un alto nivel, los entregables y artefactos del proyecto.

Análisis foda de fábricas de testing y software

El propósito de este análisis fue la de conocer como las fábricas de testing y de desarrollo de software manejan su ciclo de pruebas, de manera que logremos obtener un punto de partida en cuanto a la certificación de calidad de pruebas no funcionales. Para ello se entrevistó a los líderes de QA quienes me orientaron en su proceso principal de pruebas funcionales y no funcionales para las aplicaciones Web y Móviles. Cada organización mencionó que no se podría brindar más información debido a la confidencialidad a la que se rigen. Para mayor detalle, consultar el **Anexo 1 – Análisis FODA – Fábrica de Testing y software**.

Benchmarking de herramientas para pruebas no funcionales de aplicaciones web y móviles

Aplicaciones web

Una aplicación web es una herramienta desarrollada en lenguajes como lo son HTML o JavaScripts, la cual es publicada en un servidor web. Esta se caracteriza por la facilidad que tiene el usuario a acceder a través de un navegador web como Chrome, Firefox, Safari. Para mayor detalle consultar el **Benchmarking de herramientas web**.

Requerimientos no funcionales de aplicaciones web

Teniendo como referencia el marco de trabajo ISTQB, SWEBOK y atributos de calidad brindado por el SEI, se definieron para el modelo 6 requerimientos no funcionales y 20 sub categorías de estos.

Tabla 2: Clasificación de requerimientos no funcionales para app web

Requerimiento no funcional	Característica
Usabilidad	Navegabilidad
	Accesibilidad
Movilidad	Diseño de respuesta
Performance	Tiempo de Respuesta

	Carga
	Stress
	Volumen
Seguridad	Inyección
	Autenticación
	Secuencia de Comandos en Sitios Cruzados (XSS)
	Referencia Directa a Objetos
	Configuración de Seguridad
	Exposición de datos sensibles
	Control de Acceso a Funciones
	Peticiones en Sitios Cruzados (CSRF)
	Utilización de componentes con vulnerabilidades conocidas
	Redirecciones y reenvíos no válidos
Escalabilidad	Escalabilidad
Monitoreabilidad	Seguimiento de la aplicación en producción
	Errores

Usabilidad

Tabla 3: Subcategoría de RNF Usabilidad App Web

Subcategoría	Descripción
--------------	-------------

Navegabilidad	Esta característica se basa en permitir que el usuario tenga conocimiento de la ubicación en donde se encuentra mientras navega en la app web.
Accesibilidad	Permite que la app web sea accesible para cualquier tipo de usuario que pueda navegar.

Movilidad

Tabla 4: Subcategoría de RNF Movilidad app Web

Subcategoría	Descripción
Diseño de respuesta	La aplicación debe visualizarse de manera correcta en diferentes resoluciones de dispositivos. Además, debe funcionar de manera correcta en cada uno de estos.

Performance

Tabla 5: Subcategoría de RNF Performance app Web

Subcategoría	Descripción
Tiempo de Respuesta	Las páginas de la aplicación deben responder en un máximo de tiempo establecido.
Carga	La aplicación debe responder de manera correcta cuando el número de usuarios esperados realicen peticiones al mismo tiempo.
Stress	La aplicación debe responder de manera correcta ante la variación de peticiones, cuando se supere las peticiones esperadas, ante cargas pico.
Volumen	La aplicación debe soportar la cantidad de volumen de datos solicitada como mínima.

Seguridad

Tabla 6: Subcategoría de RNF Seguridad app Web

Subcategoría	Descripción de la característica
Inyección	Las consultas que se reciban deben prever la existencia de datos no confiables que pueden engañar al intérprete en ejecutar comandos no intencionados o acceder a datos no autorizados.
Autenticación	Las funciones de la aplicación relacionadas a autenticación y gestión de sesiones deben ser implementadas de forma correcta, para no permitir a los atacantes comprometer contraseñas, claves, token de sesiones, o explotar otras fallas de implementación para asumir la identidad de otros usuarios.
Secuencia de Comandos en Sitios Cruzados (XSS)	La aplicación debe validar y codificar de forma apropiada todos los datos que envíe al navegador. Para evitar ejecutar secuencias de comandos en el navegador de la víctima, los cuales pueden secuestrar las sesiones de usuario, destruir sitios web, o dirigir al usuario hacia un sitio malicioso.
Referencia Directa a Objetos	No se debe exponer una referencia a un objeto de implementación interno, tal como un fichero, directorio, o base de datos. Sin un chequeo de control de acceso u otra protección, los atacantes pueden manipular estas referencias para acceder datos no autorizados.
Configuración de Seguridad	Se debe contar con una buena seguridad, la cual requiere tener definida e implementada una configuración segura para la aplicación, marcos de trabajo, servidor de aplicación, servidor web, base de datos, y plataforma. Todas estas configuraciones deben ser definidas, implementadas, y mantenidas ya que por lo general no son seguras por defecto. Esto incluye mantener todo el software actualizado, incluidas las librerías de código utilizadas por la aplicación.

Exposición de datos sensibles	Los datos sensibles deben contar con métodos de protección adicionales tales como el cifrado de datos, así como también de precauciones especiales en un intercambio de datos con el navegador. Los atacantes pueden robar o modificar tales datos para llevar a cabo fraudes, robos de identidad u otros delitos.
Control de Acceso a Funciones	Se debe verificar el control de acceso cuando se accede a cada función, tanto a nivel de aplicación como del servidor. Si las solicitudes de acceso no se verifican, los atacantes podrán realizar peticiones sin la autorización apropiada.
Peticiones en Sitios Cruzados (CSRF)	La aplicación web no debe ser vulnerable ante los ataque CSRF, que obliga al navegador de una víctima autenticada a enviar una petición HTTP falsificado, incluyendo la sesión del usuario y cualquier otra información de autenticación incluida automáticamente. Esto permite al atacante forzar al navegador de la víctima para generar pedidos que la aplicación vulnerable piense son peticiones legítimas provenientes de la víctima.
Utilización de componentes con vulnerabilidades conocidas	Los componentes que utilizan las aplicaciones web tales como las librerías, los frameworks y otros módulos de software no deben ser vulnerables. Si se ataca un componente vulnerable esto podría facilitar la intrusión en el servidor o una pérdida seria de datos.
Redirecciones y reenvíos no válidos	Las aplicaciones web deben utilizar datos confiables para determinar la página de destino al redirigir y reenviar a los usuarios hacia otras páginas o sitios web. Sin una validación apropiada, los atacantes pueden redirigir a las víctimas hacia sitios de phishing o malware, o utilizar reenvíos para acceder páginas no autorizadas.

Escalabilidad

Tabla 7: Subcategorías de RNF Escalabilidad app Web

Subcategoría	Descripción de la característica
--------------	----------------------------------

Escalabilidad	La aplicación se debe construir pensando en que esta puede crecer, tanto de manera horizontal como vertical. La escalabilidad horizontal o extensión, se refiere a la adición de recursos a la unidad lógica. La escalabilidad vertical, se refiere a la adición de recursos a la unidad física.
---------------	--

Monitoreabilidad

Tabla 8: Subcategorías de RNF Monitoreabilidad app Web

Subcategoría	Descripción de la característica
Seguimiento de la aplicación en producción	Se debe medir y analizar el tráfico de la aplicación que se encuentra en producción.
Errores	Se debe realizar el monitoreo de la aplicación en funcionamiento para capturar los errores presentados y realizar el seguimiento de los mismos.

Listado de herramientas propuestas para pruebas a app web

Las herramientas para pruebas de aplicaciones Web contempladas para apoyar el modelo, ya sean open source o con licencia, se mostrarán en el siguiente cuadro.

Tabla 9: Herramientas de testing para app web

Herramienta	Descripción	Tipo Herramienta
<i>Browser Stack</i>	Tipo de Prueba: Funcional, No Funcional Tipo de Ejecución: Manual, Automatizadas	Online
<i>CrossBrowserTesting</i>	Tipo de Prueba: Funcional, No Funcional Tipo de Ejecución: Manual, Automatizadas	Online
<i>Browserling</i>	Tipo de Prueba: Funcional, No Funcional Tipo de Ejecución: Manual	Online

<i>UserTesting</i>	Tipo de Prueba: Funcional, No Funcional Tipo de Ejecución: Manual, Automatizadas	Online
<i>Device AnyWhere</i>	Tipo de Prueba: Funcional, No Funcional Tipo de Ejecución: Manual, Automatizadas	Local
<i>Testingbot</i>	Tipo de Prueba: Funcional, No Funcional Tipo de Ejecución: Manual, Automatizadas	Online
<i>Sauce labs</i>	Tipo de Prueba: Funcional, No Funcional Tipo de Ejecución: Manual, Automatizadas	Online
<i>achecker</i>	Tipo de Prueba: No Funcional Tipo de Ejecución: Manual, Automatizadas	Online
<i>validator.w3.org</i>	Tipo de Prueba: Funcional, Tipo de Ejecución: Manual, Automatizadas	Online
<i>508checker</i>	Tipo de Prueba: Funcional, No Funcional Tipo de Ejecución: Automatizadas	Online
<i>Wave</i>	Tipo de Prueba: Funcional, Tipo de Ejecución: Manual, Automatizadas	Online
<i>Taw</i>	Tipo de Prueba: Funcional, Tipo de Ejecución: Automatizadas	Online
<i>Functional Accessibility Evaluator</i>	Tipo de Prueba: Funcional, Tipo de Ejecución: Manual, Automatizadas	Online
<i>pingdom</i>	Tipo de Prueba: Funcional, No Funcional Tipo de Ejecución: Manual, Automatizadas	Online
<i>cynthiasays</i>	Tipo de Prueba: Funcional, No Funcional Tipo de Ejecución: Manual, Automatizadas	Online

<i>Soasta CloudTest</i>	Tipo de Prueba: Funcional, No Funcional Tipo de Ejecución: Automatizadas	Local
<i>Selenium IDE</i>	Tipo de Prueba: No Funcional Tipo de Ejecución: Automatizadas	Local
<i>Selenium WebDriver</i>	Tipo de Prueba: Funcional, No Funcional Tipo de Ejecución: Automatizadas	Local
<i>Ranorex</i>	Tipo de Prueba: No Funcional Tipo de Ejecución: Automatizadas	Local
<i>MobileTest.me</i>	Tipo de Prueba: No Funcional Tipo de Ejecución: Automatizadas	Online
<i>Test Complete</i>	Tipo de Prueba: No Funcional Tipo de Ejecución: Automatizadas	Online
<i>HP Functional Testing</i>	Tipo de Prueba: No Funcional Tipo de Ejecución: Automatizadas	Local
<i>Testdroid</i>	Tipo de Prueba: No Funcional Tipo de Ejecución: Automatizadas	Online
<i>Remote TestKit</i>	Tipo de Prueba: No Funcional Tipo de Ejecución: Automatizadas	Online
<i>Jmeter</i>	Tipo de Prueba: No Funcional Tipo de Ejecución: Automatizadas	Local
<i>AgileLoad</i>	Tipo de Prueba: No Funcional Tipo de Ejecución: Automatizadas	Local
<i>Load Impact</i>	Tipo de Prueba: No Funcional Tipo de Ejecución: Automatizadas	Online
<i>StresStimulus</i>	Tipo de Prueba: No Funcional Tipo de Ejecución: Automatizadas	Local

<i>Nessus</i>	Tipo de Prueba: No Funcional Tipo de Ejecución: Automatizadas	Online
<i>Fiddler</i>	Tipo de Prueba: No Funcional Tipo de Ejecución: Automatizadas	Local
<i>Rational Performance Tester</i>	Tipo de Prueba: No Funcional Tipo de Ejecución: Automatizadas	Local
<i>OWASP Zap</i>	Tipo de Prueba: No Funcional Tipo de Ejecución: Automatizadas	Local
<i>Acunetix Web Vulnerability Scanner</i>	Tipo de Prueba: No Funcional Tipo de Ejecución: Automatizadas	Local
<i>IBM Rational AppScan</i>	Tipo de Prueba: No Funcional Tipo de Ejecución: Automatizadas	Local
<i>Dynatrace</i>	Tipo de Prueba: No Funcional Tipo de Ejecución: Automatizadas	Online
<i>Raygun</i>	Tipo de Prueba: No Funcional Tipo de Ejecución: Automatizadas	Online
<i>Sentry</i>	Tipo de Prueba: No Funcional Tipo de Ejecución: Automatizadas	Online
<i>Airbrake</i>	Tipo de Prueba: No Funcional Tipo de Ejecución: Automatizadas	Online
<i>SonarQube</i>	Tipo de Prueba: No Funcional Tipo de Ejecución: Automatizadas	Online
<i>Google CodePro Analytix</i>	Tipo de Prueba: No Funcional Tipo de Ejecución: Automatizadas	Local

Listado de herramientas seleccionadas para pruebas a app web

Las herramientas con licencia consideradas son:

Ilustración 9: Herramientas seleccionadas con licencia para pruebas en web

Las herramientas open source consideradas son:

Ilustración 10: Herramientas seleccionadas open source para pruebas en web

Aplicaciones móviles

Herramienta diseñada exclusivamente para los smartphones, tablets u otros tipos de dispositivos móviles, con la finalidad de ayuda al usuario a realizar una tarea en específica de cualquier índole. Estas herramientas son desarrolladas en los distintos lenguajes compatibles actualmente en el mercado, como lo son Android, Apple y Microsoft Windows Phone. Para mayor detalle consultar el **Anexo 3 – Benchmarking de herramientas móviles**

Requerimientos no funcionales de aplicaciones móviles

Teniendo como referencia el marco de trabajo ISTQB, SWEBOK, OWASP y atributos de calidad brindado por el SEI, se definieron para el modelo 8 requerimientos no funcionales y 33 sub categorías de estos.

Tabla 10: Clasificación de requerimientos no funcionales para app móvil

Requerimiento no funcional	Sub Categoría
Usabilidad	Densidades de pantalla
	Orientación
	Back-stack/Navegabilidad
Conectividad	Conexión NG
	Conexión Wi-Fi
	Modo Airplane
Data	Ajustes de tiempo
	Internacionalización
Software	Instalación
	Actualización
	Backgrounding

	Memoria Interna
	Teclado virtual
Interrupciones Móviles	Llamadas
	Mensaje de Texto
	Notificación
	Abrir URL en la aplicación
Performance	Consumo de Batería
	Consumo de memoria
	Consumo de CPU
	Carga
	Tiempo de Respuesta
	Stress
	Volumen
Seguridad	Uso de Plataforma
	Almacenamiento de datos
	Sistema de comunicación
	Autenticación
	Uso de encriptamiento

	Autorización
	Calidad del Código del Cliente
Monitoreabilidad	Reporte de Crash
	Seguimiento del uso de la app en producción

Usabilidad

Tabla 11: Subcategorías de RNF Usabilidad app móvil

Característica	Descripción de la característica
Densidades de pantalla	La interfaz de la aplicación debe mostrarse correctamente en los dispositivos con diferentes densidades.
Orientación	La aplicación se debe mostrar correctamente en las distintas orientaciones del dispositivo, ya sea portrait (vertical) y landscape (horizontal).
Back-stack/Navegabilidad	La aplicación debe permitir realizar la navegación hacia atrás con el botón back del dispositivo.

Conectividad

Tabla 12: Subcategorías de RNF Conectividad app móvil

Característica	Descripción de la característica
Ajustes de tiempo	La aplicación debe permitir realizar la configuración de la hora y que este muestre las zonas horarias correctamente según se haya elegido.

Internacionalización	La aplicación debe ser desarrollada con los principios de internacionalización: que soporte lenguajes, que los campos sean adaptables a la longitud de las palabras traducidas, los textos no se encuentren en imágenes.
----------------------	--

Data

Tabla 13: Sub categorías de RNF Data app móvil

Característica	Descripción de la característica
Ajustes de tiempo	La aplicación debe permitir realizar la configuración de la hora y que este muestre las zonas horarias correctamente según se haya elegido.
Internacionalización	La aplicación debe ser desarrollada con los principios de internacionalización: que soporte lenguajes, que los campos sean adaptables a la longitud de las palabras traducidas, los textos no se encuentren en imágenes.

Software

Tabla 14: Subcategorías de RNF Software app móvil

Característica	Descripción de la característica
Instalación	La instalación, desinstalación y reinstalación deben realizarse correctamente.
Actualización	La actualización y actualizaciones forzadas de la aplicación se deben realizar correctamente.
Backgrounding	Cuando la aplicación pase a foreground nuevamente, esta se muestre en el mismo punto donde se encontraba antes de ser minimizada y funcione correctamente.
Memoria Interna	La aplicación debe ser ejecutada con diferentes cantidades de memoria interna del dispositivo.

Teclado virtual	El teclado virtual se debe mostrar en los momentos oportunos dentro de la aplicación.
-----------------	---

Interrupciones Móviles

Tabla 15: Subcategorías de RNF interrupciones móviles app móvil

Característica	Descripción de la característica
Llamadas	El usuario debe poder rechazar, recibir y establecer una llamada entrante mientras se encuentre ejecutando la aplicación y cuando se culmine esta llamada retorne al punto donde se encontraba en la aplicación.
Mensaje de Texto	Las notificaciones de mensajes se deben mostrar mientras se esté ejecutando la aplicación, además luego de ser leído deben retornar al mismo punto donde se encontraba en la aplicación.
Notificación	Las notificaciones de los calendarios y las alarmas no deben afectar el funcionamiento de la aplicación.
Abrir URL en la aplicación	La aplicación debe permitir abrir una url en el navegador del dispositivo desde la aplicación que se encuentre en ejecución.

Performance

Tabla 16: Subcategorías de RNF Performance app móvil

Característica	Descripción de la característica
Consumo de Batería	La aplicación no debe consumir demasiada batería al ser ejecutada en el dispositivo.

Consumo de memoria	La aplicación no debe consumir demasiada memoria al ser ejecutada.
Consumo de CPU	La aplicación debe consumir el CPU de manera óptima.
Carga	La aplicación debe responder de manera correcta cuando el número de usuarios esperados realicen peticiones al mismo tiempo.
Tiempo de Respuesta	La aplicación debe responder ante una petición en un tiempo aceptable (establecido). Servicio como interfaz.
Stress	La aplicación debe responder de manera correcta ante la variación de peticiones, cuando se supere las peticiones esperadas, ante cargas pico.
Volumen	La aplicación debe soportar la cantidad de volumen de datos solicitada como mínima.

Seguridad

Tabla 17: Subcategorías de RNF Seguridad app móvil

Característica	Descripción de la característica
Uso de Plataforma	La aplicación debe cubrir de manera adecuada los controles de seguridad proporcionados por cada plataforma, por ejemplo: Touch Id, the Keychain.
Almacenamiento de datos	La aplicación debe prevenir las amenazas de un adversario que ha alcanzado un dispositivo móvil perdido o robado, malware o cualquier aplicación que actúe en nombre del adversario y se ejecuta en el dispositivo móvil de la víctima.
Sistema de comunicación	La aplicación debe proporcionar seguridad entre los protocolos de comunicación, versiones de SSL, comunicación de activos sensibles sin cifrar.

Autenticación	La aplicación debe prevenir los ataques que aprovechan las vulnerabilidades de autenticación, los cuales suelen hacerse a través de herramientas automatizadas ya disponibles o desarrolladas a medida.
Uso de encriptamiento	La aplicación debe realizar el encriptado a los datos sensibles de los activos.
Autorización	La aplicación debe capturar cualquier fallo en la autorización (por ejemplo, las decisiones de autorización en el lado del cliente, navegación forzada, etc.). Es distinto de los problemas de autenticación (por ejemplo, el registro del dispositivo, la identificación del usuario, etc.).
Calidad del Código del Cliente	La aplicación debe prevenir los problemas de implementación a nivel de código en el cliente móvil. Esto podría capturar cosas como desbordamiento de búfer, vulnerabilidades de formato de cadena, y varios otros errores a nivel de código.

Monitoreabilidad

Tabla 18: Subcategorías de RNF Monitoreabilidad app móvil

Característica	Descripción de la característica
Reporte de Crash	La aplicación debe proporcionar un reporte de las razones por las cuales dejó de responder o sufrió una interrupción.
Seguimiento del uso de la app en producción	Se debe medir y analizar el tráfico de la aplicación que se encuentra en producción.

Listado de herramientas propuestas para pruebas a app móviles

Las herramientas para pruebas de aplicaciones móviles contempladas para apoyar el modelo, ya sean open source o con licencia, se mostrarán en el siguiente cuadro.

Tabla 19: Herramientas de testing para app móviles

Herramienta	Descripción	Tipo Herramienta
<i>Perfecto Mobile</i>	Tipo de prueba: Funcional, No Funcional Tipo de ejecución: Manual, Automatizadas	Online
<i>AWS Device Farm</i>	Tipo de prueba: Funcional, No Funcional Tipo de ejecución: Manual, Automatizadas	Online
<i>Device AnyWhere</i>	Tipo de prueba: Funcional, No Funcional Tipo de ejecución: Manual, Automatizadas	Local
<i>Sauce labs</i>	Tipo de prueba: Funcional, No Funcional Tipo de ejecución: Automatizadas	Online
<i>Xamarin Test Cloud</i>	Tipo de prueba: Funcional, No Funcional Tipo de ejecución: Automatizadas	Online
<i>Soasta TouchTest</i>	Tipo de prueba: Funcional, No Funcional Tipo de ejecución: Manual, Automatizadas	Online
<i>Soasta CloudTest</i>	Tipo de prueba: No Funcional Tipo de ejecución: Automatizadas	Local
<i>Soasta mPulse</i>	Tipo de prueba: No Funcional Tipo de ejecución: Automatizadas	Online
<i>MobileLabs</i>	Tipo de prueba: Funcional, No Funcional Tipo de ejecución: Manual Automatizadas	Online
<i>Telerik Test Studio</i>	Tipo de prueba: Funcional, No Funcional Tipo de ejecución: Manual, Automatizadas	Local

<i>Ranorex</i>	Tipo de prueba: Funcional, No Funcional Tipo de ejecución: Manual, Automatizadas	Local
<i>Silk Mobile</i>	Tipo de prueba: Funcional, No Funcional Tipo de ejecución: Automatizadas	Local
<i>Silk Performer</i> <i>CloudBurst</i>	Tipo de prueba: No Funcional Tipo de ejecución: Automatizadas	Online
<i>TestComplete</i>	Tipo de prueba: Funcional, No Funcional Tipo de ejecución: Manual, Automatizadas	Online
<i>LoadComplete</i> <i>Mobile</i>	Tipo de prueba: No Funcional Tipo de ejecución: Automatizadas	Local
<i>Jmeter</i>	Tipo de prueba: Funcional, No Funcional Tipo de ejecución: Automatizadas	Local
<i>Apica</i>	Tipo de prueba: No Funcional Tipo de ejecución: Automatizadas	Online
<i>Load Impact</i>	Tipo de prueba: No Funcional Tipo de ejecución: Automatizadas	Online
<i>Testdroid Cloud</i>	Tipo de prueba: Funcional, No Funcional Tipo de ejecución: Manual, Automatizadas	Online
<i>Crashlytics</i>	Tipo de prueba: No Funcional Tipo de ejecución: Automatizadas	Local
<i>See Test Manual</i>	Tipo de prueba: Funcional, No Funcional Tipo de ejecución: Manual	Local
<i>See Test</i> <i>Automation</i>	Tipo de prueba: Funcional, No Funcional Tipo de ejecución: Manual, Automatizadas	Local
<i>See Test Network</i>	Tipo de prueba: No Funcional Tipo de ejecución: Manual	Online

<i>TestObject</i>	Tipo de prueba: Funcional, No Funcional Tipo de ejecución: Manual, Automatizadas	Online
<i>eggPlant Mobile</i>	Tipo de prueba: Funcional Tipo de ejecución: Automatizadas	Local
<i>eggPlant Network</i>	Tipo de prueba: No Funcional Tipo de ejecución: Manual	Local
<i>Scirocco Cloud</i>	Tipo de prueba: Funcional, No Funcional Tipo de ejecución: Manual, Automatizadas	Online
<i>IBM Rational Test Workbench</i>	Tipo de prueba: Funcional, No Funcional Tipo de ejecución: Automatizadas	Local
<i>IBM Security Appscan</i>	Tipo de prueba: No Funcional Tipo de ejecución: Automatizadas	Local
<i>Acunetix WVS</i>	Tipo de prueba: No Funcional Tipo de ejecución: Automatizadas	Local
<i>HP Fortify</i>	Tipo de prueba: No Funcional Tipo de ejecución: Manual, Automatizadas	Local
<i>Fiddler</i>	Tipo de prueba: No Funcional Tipo de ejecución: Automatizadas	Local
<i>HP Network Virtualization (NV)</i>	Tipo de prueba: No Funcional Tipo de ejecución: Manual	Local
<i>Robotium</i>	Tipo de prueba: Funcional Tipo de ejecución: Automatizadas	Local
<i>Appium</i>	Tipo de prueba: Funcional, No Funcional Tipo de ejecución: Automatizadas	Local
<i>Calabash</i>	Tipo de prueba: Funcional Tipo de ejecución: Automatizadas	Online

<i>Selendroid</i>	Tipo de prueba: Funcional Tipo de ejecución: Automatizadas	Local
<i>Genymotion</i>	Tipo de prueba: Funcional, No Funcional Tipo de ejecución: Manual	Local
<i>TestFairy</i>	Tipo de prueba: Funcional, No Funcional Tipo de ejecución: Manual	Online
<i>User Testing</i>	Tipo de prueba: Funcional, No Funcional Tipo de ejecución: Manual	Online
<i>Appsee</i>	Tipo de prueba: Funcional, No Funcional Tipo de ejecución: Manual	Online
<i>UberTesters</i>	Tipo de prueba: Funcional, No Funcional Tipo de ejecución: Manual	Online
<i>AppTestNow</i>	Tipo de prueba: Funcional, No Funcional Tipo de ejecución: Manual	Online
<i>Testmunk</i>	Tipo de prueba: Funcional, No Funcional Tipo de ejecución: Manual	Online
<i>SonarQube</i>	Tipo de prueba: No Funcional Tipo de ejecución: Automatizadas	Online
<i>PMD</i>	Tipo de prueba: No Funcional Tipo de ejecución: Automatizadas	Local
<i>CodePro AnalytiX</i>	Tipo de prueba: No Funcional Tipo de ejecución: Automatizadas	Local
<i>ReSharper</i>	Tipo de prueba: No Funcional Tipo de ejecución: Automatizadas	Local

Listado de herramientas seleccionadas para pruebas a app móviles

Las herramientas con licencia consideradas son:

Ilustración 11: Herramientas seleccionadas con licencia para pruebas en web

Las herramientas open source consideradas son:

Ilustración 12: Herramientas seleccionadas open source para pruebas en web

Modelo de certificación de Pruebas No Funcionales

En base las metodologías, estándares y buenas prácticas internacionales, se ha diseñado el proceso del modelo de certificación de requerimientos no funcionales centrándose en aplicaciones web y móviles.

El proceso de certificación se encuentra compuesto por tres sub procesos:

Gestión Organizacional del Proceso de Pruebas

Capa donde se gestionan las políticas y estrategias organizacionales que seguirá el proceso de Certificación, de manera que se logre la calidad esperada bajo lineamientos estandarizados y buenas prácticas organizacionales.

Asimismo, se definieron políticas organizacionales las cuales se encuentran detalladas en el Anexo 2 – QSCERT – Política Organizacional de Pruebas.

Gestión del Proceso de Pruebas

Capa que se encarga de gestionar el proceso de pruebas, se define el Diseño de Casos de Pruebas y el Plan de Pruebas que seguirá las pruebas dinámicas. Asimismo, Realizará el control y monitoreo de las pruebas, y también de dar por finalizado un ciclo de prueba para la certificación.

Definición de roles y Responsabilidades

Para el proceso de Pruebas Propuesto, se definieron los siguientes roles y responsabilidades:

- **Jefe de Proyecto:** Persona encargada de emitir la solicitud de certificación en pruebas no funcionales, asimismo de hacer entrega de la especificación del requerimiento para que sea adicionada a la cola de pruebas actual.
- **Asistente técnico de Pruebas Senior:** Persona encargada de gestionar las pruebas según los artefactos presentados, además, es quien registra en el cronograma de pruebas la solicitud entrante. Asimismo, es quien realiza las aprobaciones respectivas durante la gestión de pruebas.
- **Asistente técnico de Pruebas:** Persona encargada de transcribir lo especificado por el asistente técnico de Pruebas, así como de diseñar los casos de prueba y el plan de pruebas para que inicie el ciclo de certificación.

- **Tester:** Personas encargada de ejecutar las pruebas según el plan de pruebas generado, usando las herramientas propuestas.

Gestión de Procesos de Pruebas Dinámicas

Capa que se encarga de ejecutar las pruebas establecidas en la capa anterior, de manera que le logre completar, mediante un cronograma de pruebas, las pruebas establecidas.

Diseño y Caracterización del Modelo de Certificación de Pruebas No Funcionales

El modelo de certificación diseñado fue elaborado bajo metodologías, estándares y buenas prácticas internacionales, de manera que se logró contemplar toda actividad correspondiente a la gestión y ejecución de pruebas, además de las actividades organizacionales las cuales son actualizadas luego de cada ejecución de prueba de manera que las políticas y estrategias organizacionales sean cada vez más sólida. Este modelo tiene la cualidad de ser flexible y modificable frente a cualquier ampliación de alcance, ya que logra cumplir con todos los requisitos mínimos para el proceso de pruebas. Para más detalle del proceso de certificación, consultar el **Anexo 3 – Diseño y Caracterización del Modelo de Certificación**. A continuación, el modelo de Certificación de Pruebas no Funcionales.

Modelo de Certificación de Pruebas No Funcionales

Ilustración 13: Modelo de certificación de Pruebas No Funcionales

Tabla 20: Modelo de certificación de Pruebas No Funcionales

N o	ENTRADA	ACTIVIDA D	SALIDA	DESCRIPCIÓN	RESPONSAB LE
-	-	Inicio	Gestión organizacional del proceso de Pruebas	Inicia la de gestión organizacional de IT Service para el proceso de Pruebas	Área de Pruebas
1.	Se requiere definir los estándares organizacionales para el proceso de pruebas	Gestión organizacional del Proceso de Pruebas	Se obtuvieron los estrategias y políticas organizacionales para el proceso de Pruebas	Se realiza la definición de políticas y estándares que adoptará el proceso de pruebas de IT Service	Área de Pruebas
	Retroalimentación del Proceso de Pruebas ejecutado				
2.	Se obtuvieron lineamientos y políticas organizacionales	Gestión del Proceso de Pruebas	Reporte de Preparación, Monitoreo y Control de Pruebas.	Se realiza toda la gestión del proceso de pruebas, de manera que el recurso pueda ejecutar sin problemas sus actividades.	Área de Pruebas
	Reporte del Proceso de Pruebas Dinámicas		Informe de Gestión de Pruebas realizada		

3.	Reporte de Preparación, Monitoreo y Control de Pruebas	Gestión del Proceso de Pruebas Dinámicas	Reporte de Finales de Pruebas Dinámicas	El recurso de IT Service ejecuta el proceso de Pruebas Dinámicas siguiente lo declarado en la Gestión del Proceso de Pruebas.	Área de Pruebas
-	Reporte Finales de Pruebas Dinámicas	Fin	—	Fin del Proceso de Pruebas	

Gestión Organizacional del Proceso de Pruebas

Ilustración 14: Gestión organizacional del proceso de pruebas

Tabla 21: Gestión Organizacional del Proceso de Pruebas

Nº	ENTRADA	ACTIVIDAD	SALIDA	DESCRIPCIÓN	RESPONSABLE

-	-	Inicio	Se requiere definir las políticas y estrategias organizacionales para el proceso de pruebas	Inicia el Proceso de Definición de Políticas y Estrategias organizacionales.	Asistente Técnico de Pruebas
1.	Se requiere definir las políticas y estrategias organizacionales	Definir Políticas de Pruebas Organizacional es	Políticas Organizacionales definidas	Se definen las políticas que adoptará el proceso de Pruebas.	Asistente Técnico de Pruebas
2.	Políticas Organizacionales definidas	Definir estrategia de Pruebas Organizacional es	Estrategias Organizacionales definidas	Se definen las estrategias que seguirá el proceso de Pruebas.	Asistente Técnico de Pruebas
3.	Propuesta Políticas y Estrategias Organizacionales	Analizar la propuesta de políticas y estrategias de Pruebas Organizacional es de pruebas	Evaluación de Políticas y Estrategias Organizacionales	El profesor Gerente evalúa las propuestas definidas por el Asistente Técnico	Analista Técnico de Pruebas Senior
4.	Evaluación de Políticas y Estrategias Organizacionales	¿Aprueba la Propuesta Presentada?	Retorna las propuestas del Asistente de Gerencia	Se evalúa si la propuesta generada por el Asistente va de acuerdo a las políticas.	Analista Técnico de Pruebas Senior
			Aprueba las Políticas y		

			Estrategias Organizacionales		
5.	Políticas y Estrategias Organizacionales Aprobadas	Publicar las políticas y estrategias de pruebas	Políticas y Estrategias Organizacionales Publicadas	Se publican las Políticas y Estrategias organizacionales para que todas las empresas puedan tener en cuenta.	Asistente Técnico de Pruebas
6.	Políticas y Estrategias Organizacionales Publicadas	Distribuir políticas de pruebas en empresas virtuales	Políticas y Estrategias Organizacionales Distribuidas	Se comunica a las empresas virtuales de las nuevas políticas y estrategias de Pruebas	Asistente Técnico de Pruebas
-	Políticas y Estrategias Organizacionales Distribuidas	Fin	-	Se finaliza el Proceso de Definición de Políticas y Estrategias de Pruebas	Asistente Técnico de Pruebas

Gestión del Proceso de Pruebas

Ilustración 15: Gestión del proceso de prueba

Tabla 22: Gestión del proceso de prueba

Nº	ENTRADA	ACTIVIDAD	SALIDA	DESCRIPCIÓN	RESPONSABLE
-	Políticas y Estrategias Organizacionales Definidas	Inicio	Se requiere gestionar las solicitudes de Pruebas	Inicia el Proceso de Planificación de Pruebas	Área de Pruebas
1.	Políticas y Estrategias Organizacionales Definidas	Planificación	Solicitudes de Pruebas Aprobadas	Se gestiona la solicitud de Pruebas enviadas por los jefes del Proyecto	Área de Pruebas
2.	Solicitudes de Pruebas Aprobadas	Control y Seguimiento	Planes de Pruebas Definidas y Reportes de Pruebas	Se realiza el monitoreo de Pruebas, obteniendo reportes de Pruebas, controlando el riesgo	Área de Pruebas

3.	Planes de Pruebas Definidas y Reportes de Pruebas	Finalización	Informe de Finalización	Se realiza la limpieza del ambiente de pruebas, adicionalmente se registran las lecciones aprendidas	Área de Pruebas
-	Informe de Finalización de Pruebas	Fin	-	Se finaliza el Proceso de Pruebas	Área de Pruebas

Planificación

Ilustración 16: Planificación

Tabla 23: Planificación

Nº	ENTRADA	ACTIVIDAD	SALIDA	DESCRIPCIÓN	RESPONSABLE
-	-	Inicio	Solicitud de Prueba	Inicia el Proceso de Planificación de Pruebas	Asistente técnico de Pruebas

1.	Solicitud de Prueba	Recibir Solicitud de Prueba	Evaluación de Solicitud de Prueba	Se recibe la solicitud de pruebas enviada por el Jefe de Proyecto	Asistente técnico de Pruebas
2.	Evaluación de Solicitud de Prueba	¿Contiene estimación y planificación documentaria?	Solicitud de Pruebas rechazada	Se evalúa la solicitud de Pruebas, se comprueba la estimación y planificación documentaria.	Asistente técnico de Pruebas
			Solicitud de Pruebas aceptada		
3.	Solicitud de pruebas rechazada	Devolver solicitud de Prueba por carencia en su presentación	Solicitud de pruebas devuelta	Se devuelve al jefe de proyecto la solicitud de pruebas debido a que no cumple con el checklist de pruebas	Asistente técnico de Pruebas
4.	Solicitud de Pruebas aceptada	Analizar contexto actual para pruebas	Contexto Definido	Se evalúa el contexto a partir de la solicitud de pruebas	Asistente técnico de Pruebas
5.	Contexto Definido	Analizar Checklist de Pruebas	Checklist de acuerdo al contexto	Se evalúa si la solicitud posee toda la documentación necesaria	Asistente técnico de Pruebas

6.	Checklist de acuerdo al contexto	Verificar solicitud con Checklist enviado	Checklist Conforme	Se verifica los documentos de la solicitud con el checklist que el Asistente Técnico	Asistente técnico de Pruebas
7.	Verificación Checklist y Solicitud de Pruebas aprobado	¿Los Documentos están conformes?	Documentos no conformes	Se evalúa si el análisis del checklist documental cumple con los requisitos	Analista Técnico de Pruebas Senior
			Documentos conformes respecto al checklist		
8.	Documentos Confirmados	Establecer cronograma de pruebas	Cronograma de pruebas actualizado	Se establecen las fechas de pruebas en el cronograma de pruebas.	Asistente técnico de Pruebas
9.	Cronograma Actualizado	Gestionar Recurso	Recurso Seleccionado	Se selecciona el recurso según los requisitos enviados por los jefes de proyecto.	Asistente técnico de Pruebas
10.	Recurso Seleccionado	Aprobar Solicitud de Prueba	Solicitud de Pruebas Aprobada	Se aprueba la solicitud luego de la evaluación realizada por el Asistente Técnico.	Analista Técnico de Pruebas Senior

11	Solicitud de Pruebas Aprobada	Actualizar cronograma de pruebas	Cronograma de pruebas Actualizado	Se actualiza el cronograma de pruebas.	Asistente técnico de Pruebas
-	Cronograma de Pruebas Actualizado	Fin	-	Finaliza el proceso de Planificación.	Asistente técnico de Pruebas

Gestionar Recurso

Ilustración 17: Gestionar recurso

Tabla 24: Gestión de Recurso

Nº	ENTRADA	ACTIVIDAD	SALIDA	DESCRIPCIÓN	RESPONSABLE
-	-	Inicio	Se necesita un recurso para realizar las pruebas de Software	Inicia el Proceso de Selección de Recursos.	Jefe de Proyecto
1.	Se necesita un recurso para realizar las	Generar Solicitud de Recurso	Solicitud de Recurso Generado	Se genera la solicitud de recurso para que pueda realizar	Jefe de Proyecto

	pruebas de Software			las pruebas respectivas.	
2.	Solicitud de Recurso Generado	Enviar Solicitud de Recurso a QS	Solicitud de Recurso enviado	Se envía la solicitud para que evalúe un recurso disponible.	Jefe de Proyecto
	Solicitud de Recursos no recibida				
3.	Solicitud de Recurso enviado	¿Llegó Solicitud?	Solicitud aceptada	Se verifica si la Solicitud de Recursos llegó a la bandeja de entrada.	Asistente Técnico de Pruebas
			Solicitud de Recursos no recibida		
4.	Solicitud de Recursos Aceptada	Evaluar Solicitud	Solicitud Evaluada	Se procede con la evaluación de la solicitud, si cumple con los requisitos	Asistente Técnico de Pruebas
5.	Solicitud Evaluada	¿Solicitud Completa?	Solicitud de Recursos Incompleta	Se evalúa si la solicitud de recursos se encuentra completa.	Asistente Técnico de Pruebas
			Solicitud de Recursos Aprobada		
6.	Solicitud de Recursos Incompleta	Completar datos faltantes	Solicitud de Recursos Actualizada	La solicitud no cumple con los requisitos mínimos.	Jefe de Proyecto

7.	Solicitud de Recursos Aprobada	Buscar Recurso disponible	Recurso Disponible	Se verifica en el registro si se encuentra un recurso disponible para asignar.	Asistente Técnico de Pruebas
8.	Recurso Disponible	¿Cumple con requisitos del Jefe de Proyecto?	Recurso no cumple con requisitos expuestos en solicitud Recurso cumple con requisitos para proceso de pruebas	Se evalúa si el recurso disponible cumple con los requisitos solicitados por el Jefe de Proyecto.	Asistente Técnico de Pruebas
9.	Recurso disponible y cumple con los requisitos definidos	Entregar lineamientos y Documentos aprobados	Recurso informado de proyecto asignado	Se entregan los lineamientos que el jefe de proyecto compartió para entregar al recurso asignado	Asistente Técnico de Pruebas
10.	Recurso informado de proyecto asignado	Avisar a Jefe de Proyecto recurso asignado	Jefe de Proyecto informado	Se confirma al Jefe de Proyecto el recurso asignado.	Asistente Técnico de Pruebas
-	Jefe de Proyecto informado	Fin	-	Finaliza el proceso de Gestión de Recursos.	Asistente Técnico de Pruebas

Control y Seguimiento

Ilustración 18: Control y Seguimiento

Tabla 25 Control y seguimiento

Nº	ENTRADA	ACTIVIDAD	SALIDA	DESCRIPCIÓN	RESPONSABLE
-	-	Inicio	Se genera la necesidad de Gestionar los planes para el Proceso de Prueba	Se inicia el Proceso de Control y Seguimiento	Jefe de Proyecto
1.	Se genera la necesidad de Gestionar los planes para el Proceso de Prueba	Gestión de Planes de Pruebas	Planes de Pruebas Generados	Se encarga de gestionar el Plan de Pruebas que usará el Recurso.	Jefe de Proyecto

2.	Planes de Pruebas Generados	Monitoreo de Pruebas	Informe de Monitoreo de Pruebas	Se encarga de llevar el seguimiento del proceso de pruebas.	Analista Técnico de Pruebas
3.	Informe de Monitoreo de Pruebas	Control y Mejora de Pruebas	Informe de Gestión de Riesgos del Proyecto	Se encarga de controlar el proceso de pruebas reconociendo los riesgos asociados y tratarlos a tiempo.	Analista Técnico de Pruebas
4.	Informe de Monitoreo de Pruebas Informe de Gestión de Riesgos del Proyecto	Reporte de las Pruebas	Informe Final del Proceso de Pruebas	Se encarga de registrar el comportamiento de las pruebas.	Analista Técnico de Pruebas
-	Informe Final del Proceso de Pruebas	Fin	-	Finaliza el proceso de Control y Seguimiento de Pruebas	Analista Técnico de Pruebas

Gestión de Planes de Pruebas

Ilustración 19: Gestión de plan de pruebas

Tabla 26 Gestión de Plan de Pruebas

Nº	ENTRADA	ACTIVIDAD	SALIDA	DESCRIPCIÓN	RESPONSABLE
-	-	Inicio	Se genera la necesidad de Gestionar Planes de Prueba.	Se inicia el Proceso de Gestión de Planes de Prueba.	Jefe de Proyecto
1.	Se genera la necesidad de Gestionar Planes de Prueba	Definición del Plan de Pruebas	Plan de Prueba Generado	Se gestiona el Plan de Pruebas de los jefes del proyecto para el proceso de pruebas.	Jefe de Proyecto
2.	Plan de Prueba Generado	Gestionar Ambiente de Pruebas	Ambiente preparado para el proceso de Pruebas	Se gestiona el ambiente donde el recurso ejecutará sus pruebas según el	Asistente Técnico de Pruebas

				Plan de Prueba desarrollado.	
-	Ambiente preparado para el proceso de Pruebas	Fin	-	Se finaliza el proceso de Gestión de Planes de Prueba	Asistente Técnico de Pruebas

Definición del Plan de Pruebas

Ilustración 20: Definición de plan de pruebas

Tabla 27: Definición de plan de pruebas

Nº	ENTRADA	ACTIVIDAD	SALIDA	DESCRIPCIÓN	RESPONSABLE
-	-	Inicio	Se genera la necesidad de establecer el Plan para el Proceso de Pruebas.	Se inicia el proceso de Definición del Plan de Pruebas	Jefe de Proyecto

1.	Se genera la necesidad de establecer el Plan para el Proceso de Pruebas.	Definir perfiles para las pruebas	Perfiles Definidos	Se realiza la definición de perfiles según funciones y actividades de acuerdo a las pruebas.	Jefe de Proyecto
2.	Perfiles Definidos	Identificar escenarios de Pruebas no Funcionales	Escenarios parcialmente Definidos	Se identifican los escenarios que se usarán en el proceso de pruebas.	Jefe de Proyecto
3.	Escenarios parcialmente Definidos	Establecer condiciones para cada escenario	Escenarios Definidos	Se definen las condiciones que las pruebas adoptarán por escenario de prueba.	Jefe de Proyecto
4.	Escenarios Definidos	Definir los casos de prueba para cada escenario	Casos de Pruebas definidos para cada escenario	Se definen los casos de pruebas que serán usados por escenario.	Jefe de Proyecto
5.	Casos de Pruebas definidos para cada escenario	¿Todos los escenarios fueron contemplados?	Escenarios Incompletos	Se evalúa si los escenarios establecidos cumplen con lo necesario para definirse en el Plan de Pruebas	Jefe de Proyecto
			Escenarios de Pruebas Definidos		

6.	Escenarios de Pruebas Definidos	Completar Casos de Prueba en Checklist de Prueba	Checklist de Prueba completo	La plantilla de Checklist para Casos de uso es completada para el uso del Recurso.	Jefe de Proyecto
7.	Checklist de Prueba completo	Establecer los criterios de Verificación e Indicadores de Aceptación para las pruebas	Criterios e Indicadores seleccionados	Se definen los criterios para las pruebas de manera que se encuentren bien alineadas.	Jefe de Proyecto
	Criterios e Indicadores insuficientes				
8.	Criterios e Indicadores seleccionados	¿Se contempla lo necesario?	Criterios e Indicadores insuficientes	Se evalúa si los criterios e indicadores son suficientes para las pruebas	Jefe de Proyecto
			Criterios e Indicadores Aprobados		
9.	Criterios e Indicadores Aprobados	Definir Recursos informáticos principales	Recursos Informáticos seleccionados	Se eligen los recursos informáticos se necesitan para las pruebas.	Jefe de Proyecto
10.	Recursos Informáticos seleccionados	Definir Alcance Final de Pruebas No Funcionales	Alcance de Pruebas Definidos	Se define el alcance que las pruebas tendrán para llevarse a cabo.	Jefe de Proyecto

11	Alcance de Pruebas Definidos	Gestionar Pruebas No Funcionales	Pruebas No Funcionales seleccionadas según Plan de Pruebas	Se realiza la gestión de los requerimientos no funcionales que intervendrán en el proceso de pruebas según alcance.	Jefe de Proyecto
12	Pruebas No Funcionales seleccionadas según Plan de Pruebas	Elaborar Plan de Prueba	Plan de Pruebas desarrollado	Se genera el Plan de Prueba con los puntos definidos.	Jefe de Proyecto
13	Plan de Pruebas desarrollado	Enviar Plan de Prueba	Plan de Pruebas enviado	Se envía el plan de Pruebas al Asistente Técnico.	Jefe de Proyecto
-	Plan de Pruebas enviado	Fin	-	Se finaliza el proceso de Definición del Plan de Pruebas	Jefe de Proyecto

Gestionar Pruebas No Funcionales

Ilustración 21: Gestión Pruebas no funcionales

Tabla 28: Gestión de pruebas no funcionales

Nº	ENTRADA	ACTIVIDAD	SALIDA	DESCRIPCIÓN	RESPONSABLE
-	-	Inicio	Se genera la necesidad de Gestionar Planes de Prueba	Se inicia el proceso de Gestionar Pruebas No Funcionales.	Jefe de Proyecto
1.	Se genera la necesidad de Gestionar Planes de Prueba	Revisar Alcance de Pruebas	Plan de Prueba Generado	Se revisa el alcance definido anteriormente para validar el tipo de aplicación.	Jefe de Proyecto
2.	Plan de Prueba Generado	Identificar tipo de aplicación	Ambiente preparado para el proceso de Pruebas	Se identifica el tipo de aplicación entrante para las pruebas	Jefe de Proyecto

3.	Ambiente preparado para el proceso de Pruebas	Validar Tipo de Aplicación	Alcance definido y tipo de aplicación detectado	Se evalúa si la aplicación es Web o móvil	Jefe de Proyecto
			Alcance definido y tipo de aplicación detectado		
4.	Alcance definido y tipo de aplicación detectado	Gestión de Pruebas no funcionales para Aplicaciones Web	Pruebas No Funcionales para aplicaciones Web gestionadas	Se desea gestionar las pruebas no funcionales para aplicaciones web	Jefe de Proyecto
5.	Alcance definido y tipo de aplicación detectado	Gestión de Pruebas No Funcionales para aplicaciones Móviles	Pruebas No Funcionales para aplicaciones Móviles gestionadas	Se desea gestionar las pruebas no funcionales para aplicaciones móviles	Jefe de Proyecto
6.	Pruebas No Funcionales para aplicaciones Web gestionadas	Pruebas Gestionadas	Pruebas No funcionales Definidas y preparadas	Se prepararon las pruebas no funcionales según el alcance definido.	Jefe de Proyecto
	Pruebas No Funcionales para aplicaciones				

	Móviles gestionadas				
7.	Pruebas No funcionales Definidas y preparadas	Actualizar Checklist de pruebas	Checklist de Pruebas Actualizado	Se actualiza el Checklist de Pruebas para preparar los casos de prueba.	Jefe de Proyecto
-	Checklist de Pruebas Actualizado	Fin	-	Finaliza el Proceso de Gestión de Pruebas No Funcionales	Jefe de Proyecto

Gestión de Pruebas No Funcionales para Aplicaciones Web

Ilustración 22: Gestión de Pruebas No Funcionales para Aplicaciones Web

Tabla 29: Gestión de Pruebas No Funcionales para Aplicaciones Web

Nº	ENTRADA	ACTIVIDAD	SALIDA	DESCRIPCIÓN	RESPONSABLE
-	-	Inicio	Se genera la necesidad de gestionar las pruebas no funcionales para aplicaciones web.	Se inicia el Proceso de Gestión de Pruebas No Funcionales para Aplicaciones Web	Jefe de Proyecto

1.	Se genera la necesidad de gestionar las pruebas no funcionales para aplicaciones web.	Seleccionar Pruebas No Funcionales	Prueba(s) No Funcionales seleccionada(s)	Según el alcance Definido anteriormente se seleccionan las pruebas no funcionales que abarcarán en las pruebas.	Jefe de Proyecto
2.	Prueba(s) No Funcionales seleccionada(s)	Gestionar Prueba de Usabilidad	Pruebas de Usabilidad Gestionada	Se gestiona la prueba de Usabilidad para las pruebas	Jefe de Proyecto
3.	Prueba(s) No Funcionales seleccionada(s)	Gestionar Prueba de Movilidad	Pruebas de Movilidad Gestionada	Se gestiona la prueba de Movilidad para las pruebas	Jefe de Proyecto
4.	Prueba(s) No Funcionales seleccionada(s)	Gestionar Prueba de Performance	Pruebas de Performance Gestionada	Se gestiona la prueba de Performance para las pruebas	Jefe de Proyecto
5.	Prueba(s) No Funcionales seleccionada(s)	Gestionar Pruebas de Escalabilidad	Pruebas de Escalabilidad Gestionada	Se gestiona la prueba de Escalabilidad para las pruebas	Jefe de Proyecto
6.	Prueba(s) No Funcionales seleccionada(s)	Gestionar Prueba de Monitoreabilidad	Pruebas de Monitoreabilidad Gestionada	Se gestiona la prueba de Monitoreabilidad para las pruebas	Jefe de Proyecto

7.	Pruebas de Usabilidad Gestionada	¿Se abarca según el alcance?	Prueba(s) No Funcionales gestionada(s)	Se evalúa si las pruebas gestionadas son las que abarcan en el alcance de pruebas.	Jefe de Proyecto
	Pruebas de Movilidad Gestionada				
	Pruebas de Performance Gestionada				
	Pruebas de Escalabilidad Gestionada				
	Pruebas de Monitoreabilidad Gestionada				
	Prueba(s) No Funcionales gestionada(s)	Fin	-	Finaliza el proceso de Gestión de Pruebas No Funcionales para Aplicaciones Web	Jefe de Proyecto

Gestión de Pruebas No Funcionales para Aplicaciones Móviles

Ilustración 23: Gestión de Pruebas No Funcionales para Aplicaciones Móviles

Tabla 30: Gestión de Pruebas No Funcionales para Aplicaciones Móviles

Nº	ENTRADA	ACTIVIDAD	SALIDA	DESCRIPCIÓN	RESPONSABLE
-	-	Inicio	Se genera la necesidad de gestionar las pruebas no funcionales para aplicaciones web.	Se inicia el Proceso de Gestión de Pruebas No Funcionales para Aplicaciones Web	Jefe de Proyecto

1.	Se genera la necesidad de gestionar las pruebas no funcionales para aplicaciones web.	Seleccionar Pruebas No Funcionales	Prueba(s) No Funcionales seleccionada(s)	Según el alcance Definido anteriormente se seleccionan las pruebas no funcionales que abarcarán en las pruebas.	Jefe de Proyecto
2.	Prueba(s) No Funcionales seleccionada(s)	Gestionar Prueba de Usabilidad	Pruebas de Usabilidad Gestionada	Se gestiona la prueba de Usabilidad para las pruebas	Jefe de Proyecto
3.	Prueba(s) No Funcionales seleccionada(s)	Gestionar Prueba de Conectividad	Pruebas de Conectividad Gestionada	Se gestiona la prueba de Conectividad para las pruebas	Jefe de Proyecto
4.	Prueba(s) No Funcionales seleccionada(s)	Gestionar Prueba de Data	Pruebas de Data Gestionada	Se gestiona la prueba de Data para las pruebas	Jefe de Proyecto
5.	Prueba(s) No Funcionales seleccionada(s)	Gestionar Pruebas de Software	Pruebas de Software Gestionada	Se gestiona la prueba de Software para las pruebas	Jefe de Proyecto
6.	Prueba(s) No Funcionales seleccionada(s)	Gestionar Prueba de Interrupciones Móviles	Pruebas de Interrupciones Móviles Gestionada	Se gestiona la prueba de Interrupciones Móviles para las pruebas	Jefe de Proyecto

7.	Prueba(s) No Funcionales seleccionada(s)	Gestionar Prueba de Performance	Pruebas de Performance Gestionada	Se gestiona la prueba de Performance para las pruebas	Jefe de Proyecto
8.	Prueba(s) No Funcionales seleccionada(s)	Gestionar Prueba de Seguridad	Pruebas de Seguridad Gestionada	Se gestiona la prueba de Seguridad para las pruebas	Jefe de Proyecto
9.	Prueba(s) No Funcionales seleccionada(s)	Gestionar Prueba de Monitoreabilidad	Pruebas de Monitoreabilidad Gestionada	Se gestiona la prueba de Monitoreabilidad para las pruebas	Jefe de Proyecto
10.	Prueba de Usabilidad Gestionada	¿Se abarca según el alcance?	Prueba(s) No Funcionales gestionada(s)	Se evalúa si las pruebas gestionadas son las que abarcan en el alcance de pruebas.	Jefe de Proyecto
	Prueba de Conectividad Gestionada				
	Prueba de Data Gestionada				
	Pruebas de Software Gestionada				
	Prueba de Interrupciones Móviles Gestionada				

	Prueba de Performance Gestionada				
	Prueba de Seguridad Gestionada				
	Prueba de Monitoreabilidad Gestionada				
-	Prueba(s) No Funcionales gestionada(s)	Fin	-	Finaliza el proceso de Gestión de Pruebas No Funcionales para Aplicaciones Web	Jefe de Proyecto

Gestionar Ambientes de Pruebas

Ilustración 24: Gestión ambiente de pruebas

Tabla 31: Gestión ambiente de pruebas

N°	ENTRADA	ACTIVIDAD	SALIDA	DESCRIPCIÓN	RESPONSABLE
-	-	Inicio	Se genera la necesidad de gestionar el ambiente para pruebas.	Inicia el proceso de Gestión de ambientes para pruebas.	Asistente Técnico de Pruebas
1.	Se genera la necesidad de gestionar el ambiente para pruebas.	Verificar Cronograma de Actividades	Cronograma de pruebas verificado	Se verifica el cronograma actual de seleccionar el proceso de pruebas más cercano.	Asistente Técnico de Pruebas
2.	Cronograma de pruebas verificado	Seleccionar Plan de Pruebas	Plan de Pruebas Seleccionado	Se selecciona el plan de Prueba según revisión del cronograma de pruebas	Asistente Técnico de Pruebas
3.	Plan de Pruebas Seleccionado	Verificar ambiente de pruebas	Ambiente de Pruebas verificado	Se verifica si el ambiente actual.	Asistente Técnico de Pruebas
4.	Ambiente de Pruebas verificado	¿Ambiente preparado?	Ambiente de Pruebas incompleto	Se evalúa si se necesita mejoras.	Asistente Técnico de Pruebas
			Ambiente de Pruebas Comprobado		

5.	Ambiente de Pruebas incompleto	Generar la solicitud de Recursos Informáticos	Solicitud de Recursos Informáticos generado	Se genera la solicitud de Recursos para enviar a DataCenter	Asistente Técnico de Pruebas
6.	Solicitud de Recursos Informáticos generado	Enviar solicitud a Asistente de Gerencia de Data Center	Solicitud de Recursos Informáticos enviado	Se envía la solicitud al Líder de Infraestructura para que evalúe.	Asistente Técnico de Pruebas
-	Solicitud de Recursos Informáticos enviado	Esperar respuesta de Asistente de IT Expert	Solicitud de Recursos Informáticos recibido	Se esperan 2 días para respuesta de Asistente Técnico.	Asistente Técnico de Pruebas
7.	Solicitud de Recursos Informáticos recibido	¿Se actualizó el ambiente según solicitud?	Ambiente de Pruebas incompleto	Se evalúa si la solicitud cumple con requisitos	Asistente Técnico de Pruebas
			Ambiente de Pruebas preparado		
8.	Ambiente de Pruebas incompleto	Informar sobre carencias del ambiente	Faltas en el ambiente de pruebas comunicado	Se reenvía solicitud incompleta	Asistente Técnico de Pruebas
9.	Ambiente de Pruebas preparado	Confirmar ambiente	Ambiente de Pruebas Confirmado	Se confirma que el ambiente se encuentra preparado	Asistente Técnico de Pruebas

10	Ambiente de Pruebas Confirmado	Actualizar Cronograma de Actividades	Cronograma de Pruebas actualizado	El cronograma de actividades se actualiza frente a próximas pruebas.	Asistente Técnico de Pruebas
-	Cronograma de Pruebas actualizado	Fin	-	Finaliza el Proceso de Gestión de ambientes para Pruebas.	Asistente Técnico de Pruebas

Monitoreo de Pruebas

Ilustración 25: Monitoreo de pruebas

Tabla 32: Monitoreo de pruebas

Nº	ENTRADA	ACTIVIDAD	SALIDA	DESCRIPCIÓN	RESPONSABLE
-	-	Inicio	Se genera la necesidad de Monitorear la Ejecución de las Pruebas	Se inicia el Proceso de Monitoreo de Pruebas	Analista Técnico de Pruebas
1.	Se genera la necesidad de Monitorear la	Verificar Cronograma de Pruebas	Cronograma de Pruebas validado	Se verifica el cronograma de pruebas actual	Analista Técnico de Pruebas

	Ejecución de las Pruebas				
2.	Cronograma de Pruebas validado	Seleccionar Proyecto a evaluar	Proyecto seleccionado para ejecución de pruebas	Se selecciona el proyecto en ejecución de pruebas.	Analista Técnico de Pruebas
3.	Proyecto seleccionado para ejecución de pruebas	Obtener Documento de Especificación de Requerimiento	Obtener Documentación del Proyecto seleccionado	Se obtiene el Documento de Especificación de Requerimiento para validar ejecución.	Analista Técnico de Pruebas
4.	Obtener Documentación del Proyecto seleccionado	Obtener Resultados Parciales de Pruebas	Informe de Pruebas realizadas	Se solicita al recurso el informe parcial de pruebas.	Analista Técnico de Pruebas
5.	Informe de Pruebas realizadas	Verificar y analizar indicadores	Indicadores Verificados	Se verifica y validan los indicadores de la ejecución de pruebas	Analista Técnico de Pruebas
6.	Indicadores Verificados	Revisar Variabilidad de Indicadores	Informe de Indicadores	Se revisa la variabilidad de indicadores durante la ejecución de pruebas	Analista Técnico de Pruebas

7.	Informe de Indicadores	Emitir informe de Pruebas	Informe de Pruebas	Se emite informe de Monitoreo de Pruebas	Analista Técnico de Pruebas
-	Informe de Pruebas	Fin	-	Se finaliza el Proceso de Monitoreo de Pruebas	Analista Técnico de Pruebas

Control y Mejora de Pruebas

Ilustración 26: Control y mejora de pruebas

Tabla 33: Control y mejora de pruebas

Nº	ENTRADA	ACTIVIDAD	SALIDA	DESCRIPCIÓN	RESPONSABLE
-	-	Inicio	Se genera la necesidad de Verificar los riesgos en las pruebas	Inicia el Proceso para Controlar Y Mejorar Las Pruebas	Analista Técnico de Pruebas
1.	Se genera la necesidad de Verificar los riesgos en las pruebas	Verificar cronograma de Pruebas	Cronograma de Pruebas validado	Se verifica el cronograma de pruebas actual	Analista Técnico de Pruebas

2.	Cronograma de Pruebas validado	Obtener los Informes de Pruebas	Informe de Pruebas obtenidos	Se obtiene el Informe parcial de pruebas	Analista Técnico de Pruebas
3.	Informe de Pruebas obtenidos	Verificar desempeño de Pruebas	Informe de Pruebas Verificado	Se evalúa el desempeño de las pruebas actual	Analista Técnico de Pruebas
4.	Informe de Pruebas Verificado	Identificar componentes afectados	Componentes Detectados	Se identifican los factores de las pruebas afectados	Analista Técnico de Pruebas
5.	Componentes Detectados	Identificar factores de afectación	Factores técnicos Identificados	Se identifican los factores	Analista Técnico de Pruebas
	Incidentes no resueltos				
6.	Factores técnicos Identificados	¿Son todos los factores a tratar?	Factores faltantes	Se evalúa si todos los factores han sido identificados	Analista Técnico de Pruebas
			Factores identificados		
7.	Factores identificados	Hacer estimación de Recuperación	Estimación definida de recuperación	Se realiza la estimación de cuanto demorará la recuperación de los incidentes	Analista Técnico de Pruebas
8.	Estimación definida de recuperación	Definir punto de Referencia	Punto de Referencia de para recuperación definido	Se define el punto de referencia de acuerdo a la recuperación	Analista Técnico de Pruebas

9.	Punto de Referencia de para recuperación definido	Registrar Riesgos detectados	Riesgos registrados	Se registran los riesgos asociados en la plataforma de incidentes	Analista Técnico de Pruebas
10.	Riesgos registrados	Implementar acciones correctivas	Acciones correctivas implementadas	Se implementa las acciones correctivas a partir de la estimación realizada.	Analista Técnico de Pruebas
11.	Acciones correctivas implementadas	¿Se resolvieron los incidentes?	Incidentes no resueltos	Se evalúa si los incidentes fueron resueltos a partir del análisis previo realizado	Analista Técnico de Pruebas
			Incidentes corregidos		
12.	Incidentes corregidos	Registrar acciones correctivas	Acciones registradas	Se registran las acciones tomadas para mitigar los incidentes	Analista Técnico de Pruebas
13.	Acciones registradas	Generar informe de Pruebas	Informe de Pruebas Generado	Se genera el informe de Pruebas en base al control realizado	Analista Técnico de Pruebas
	Informe de Pruebas Generado	Fin	-	Finaliza el Proceso de Control y	Analista Técnico de Pruebas

				Monitoreo de Pruebas	
--	--	--	--	----------------------	--

Reporte de Pruebas

Ilustración 27: Reporte de pruebas

Tabla 34: Reporte de pruebas

Nº	ENTRADA	ACTIVIDAD	SALIDA	DESCRIPCIÓN	RESPONSABLE
-	-	Inicio	Se genera la necesidad de generar los reportes de pruebas.	Se inicia el Proceso de Reporte de Pruebas.	Tester
1.	Se genera la necesidad de generar los reportes de pruebas.	Recibir informe de Pruebas	Informe de Pruebas	Se recibe el Informe de Pruebas Finales del Recurso.	Tester
2.	Informe de Pruebas	Registrar informe	Informe Actualizado	Se registran los incidentes de manera que se pueda gestionar las acciones correctivas.	Tester

3.	Informe Actualizado	Analizar registro de pruebas	Informe de Pruebas	Se analiza el registro de incidentes para tomar decisiones	Tester
4.	Informe de Pruebas	Realizar análisis de técnico	Informe de Pruebas	Se analiza la infraestructura utilizada para tomar decisiones finales	Tester
5.	Informe de Pruebas	Registrar análisis final	Informe de Pruebas revisado	Se une ambos análisis previos para generar el análisis final de las pruebas.	Tester
6.	Informe de Pruebas revisado	Generar Informe Técnico de Pruebas	Informe técnico de Pruebas generado	Se genera el informe técnico de las pruebas para informar estado actual de la infraestructura.	Tester
7.	Informe técnico de Pruebas generado	Generar Informe Final de Pruebas	Informe Final técnico de Pruebas	Se genera el informe final de pruebas.	Tester
8.	Informe Final técnico de Pruebas	Enviar Respuesta técnica	Informe Final técnico de Pruebas enviado	Se envía respuesta técnica a Asistente Técnico	Tester

9.	Informe Final técnico de Pruebas enviado	Registrar Cierre de Pruebas	Informe Final técnico de Pruebas enviado	Se cierra la prueba y se registra	Tester
-	Informe Final de Riesgos	Fin	-	Se finaliza el Proceso de Reporte de Pruebas	Tester

Finalización

Ilustración 28: Finalización

Tabla 35: Finalización

Nº	ENTRADA	ACTIVIDAD	SALIDA	DESCRIPCIÓN	RESPONSABLE
-	-	Inicio	Se genera la necesidad de iniciar el proceso de Finalización de Pruebas	Se inicia el Proceso de Finalización de Pruebas	Tester
1.	Se genera la necesidad de iniciar el proceso de	Obtener Informe Final de Pruebas	Informe Final de Pruebas obtenido	Se obtiene el Proceso Final de Pruebas generado en el	Tester

	Finalización de Pruebas			subproceso anterior.	
2.	Informe Final de Pruebas obtenido	Reconocer recursos utilizados	Recursos identificados	Se reconocen los recursos usados para saber estado	Tester
3.	Recursos identificados	Limpiar Entorno	Entorno listo pruebas	Se limpia el entorno de manera que quede listo para una prueba futura	Tester
4.	Entorno listo pruebas	Actualizar cronograma de Actividades	Cronograma de Actividades Actualizado	Se actualiza el cronograma de actividades para priorizar otras pruebas.	Tester
5.	Cronograma de Actividades Actualizado	Registrar Lecciones aprendidas	Lecciones Registradas	Se registran las lecciones para futuros procesos de pruebas.	Tester
6.	Lecciones Registradas	Realizar informe de finalización	Informe de Finalización generado	Se genera el Informe de Finalización de Pruebas	Tester
-	Informe de Finalización generado	Fin	-	Finaliza el Proceso de Pruebas	Tester

Gestión de Procesos de Pruebas Dinámicas

Ilustración 29: Gestión de pruebas dinámicas

Tabla 36: Gestión de pruebas dinámicas

Nº	ENTRADA	ACTIVIDAD	SALIDA	DESCRIPCIÓN	RESPONSABLE
-	-	Inicio	Se genera la necesidad de generar los reportes de pruebas.	Se inicia el proceso de Gestión de Pruebas Dinámicas	Tester
1.	Se genera la necesidad de generar reportes de pruebas.	Diseño e implementación de pruebas	Informe de Pruebas	Se gestiona el diseño e implementación de pruebas	Tester
2.	Informe de Pruebas	Ejecución de pruebas	Informe de Pruebas, agregado defectos	Se gestiona la ejecución de Pruebas	Tester
3.	Informe de Pruebas, agregado defectos	Gestión de Informes de Incidentes	Informe de Incidentes	Se gestionan los informes de Incidentes	Tester

-	Informe de Incidentes	Fin	-	Finaliza el Proceso de Pruebas Dinámicas	Tester
---	-----------------------	-----	---	--	--------

Diseño e Implementación de Pruebas

Ilustración 30: Diseño e implementación de pruebas

Tabla 37: Diseño e implementación de pruebas

Nº	ENTRADA	ACTIVIDAD	SALIDA	DESCRIPCIÓN	RESPONSABLE
-	-	Inicio	Se genera la necesidad de ejecutar las pruebas	Inicia el Proceso de Diseño e Implementación de Pruebas.	Tester
1.	Se genera la necesidad de ejecutar las pruebas	Obtener Plan de Pruebas	Plan de Pruebas obtenido	Se obtiene el plan de pruebas elaborado.	Tester
2.	Plan de Pruebas obtenido		Especificación de requerimiento	Se identifican los	Tester

	Requisitos incompletos	Identificar características básicas		requerimientos para llevar a cabo las pruebas.	
3.	Especificación de requerimiento	¿Se tienen todas las especificaciones del requerimiento?	<div> Especificación de requerimiento aprobado </div> <div> Necesidad de Definir requerimientos no funcionales </div>	Se valida si se tienen todos los requerimientos para las pruebas.	Tester
4.	Informe de Pruebas	Establecer condiciones para pruebas	Condiciones establecidas	Se establecen las condiciones primordiales para las pruebas	Tester
5.	Informe de Pruebas	Identificar casos de pruebas	Casos de Pruebas definidos	Se identifican los casos de pruebas para el proyecto	Tester
6.	Casos de Pruebas definidos	Definir Conjuntos de Pruebas	Conjunto de Pruebas definidos	Se definen los conjuntos de Pruebas para el proyecto	Tester
7.	Informe de Pruebas	Definir procedimientos y scripts de pruebas	Scripts y Procedimientos definidos	Procedimientos y scripts definidos y preparados	Tester
8.	Necesidad de Definir	Definir Requerimientos No Funcionales	Requerimientos No Funcionales definidos	Al no tener un documento o especificación	Tester

	Requerimientos No Funcionales			de requerimiento, se procede a definir los requerimientos No Funcionales	
9.	Informe de Pruebas	¿Se encuentran todos los requisitos para iniciar pruebas?	Informe de Pruebas rechazado Informe de Pruebas aprobado	Se evalúa si los requisitos son los suficientes para las pruebas	Tester
10.	Informe de Pruebas aprobado	Registrar Pruebas	Informe de Pruebas registrado	Se registran las pruebas en el aplicativo para que el asistente lo vea.	Tester
11.	Informe de Pruebas registrado	Actualizar Cronograma de pruebas	Cronograma actualizado	Se actualiza el cronograma de pruebas luego de preparar los requisitos de pruebas	Tester
-	Cronograma actualizado	Fin	-	Finaliza el proce	Tester

Ejecución de Pruebas

Ilustración 31: Ejecución de pruebas

Tabla 38 Ejecución de pruebas

Nº	ENTRADA	ACTIVIDAD	SALIDA	DESCRIPCIÓN	RESPONSABLE
-	-	Inicio	Se genera la necesidad de ejecutar las pruebas respectivas	Se inicia el Proceso de Ejecución de Pruebas	Tester
1.	Se genera la necesidad de ejecutar las pruebas respectivas	Obtener procedimientos y scripts definidos	Plan de Pruebas	Se obtiene los procedimientos y scripts para el proceso de pruebas	Tester
2.	Procedimientos y Scripts definidos	Obtener Plan de Pruebas	Plan de Pruebas	Se obtiene el Plan de Pruebas definido para las pruebas	Tester
3.	Plan de Pruebas	Verificar Entorno de Pruebas	Plan de Pruebas validado	Se verifica el entorno donde se	Tester

				ejecutarán las pruebas	
4.	Plan de Pruebas validado	¿Se encuentra el ambiente de pruebas listo?	Reporte de Defectos Generado	Se evalúa si los ambientes se encuentran listos	Tester
			Plan de Pruebas validado		
5.	Reporte de Defectos Generado	Elaborar reporte de defectos	Reporte de Defectos completado	Se elabora los reportes de defectos	Tester
6.	Reporte de Defectos completado	Enviar Reporte de defectos	Reporte de Defectos enviado	Se envía el reporte de defectos al Jefe de Proyecto	Tester
7.	Plan de Pruebas validado	Ejecutar scripts de pruebas	Plan de Pruebas ejecutado	Se ejecutan las pruebas con los scripts definidos	Tester
8.	Plan de Pruebas ejecutado	Comprobar contra manual de componentes	Plan de Pruebas comprobado	Se comprueba los componentes evaluados	Tester
9.	Plan de Pruebas comprobado	Validación y Verificación del proceso de Despliegue	Plan de Pruebas ejecutado	Se valida si el proceso de despliegue se ejecutó de manera normal	Tester
10.	Plan de Pruebas ejecutado	Registrar pruebas	Informe de Pruebas Registrado	Se registran las pruebas realizadas	Tester

-	Informe de Pruebas Registrado	Fin	-	Finaliza el Proceso de Ejecución de Pruebas	Tester
---	-------------------------------	-----	---	---	--------

Gestión de Informe de Incidentes

Ilustración 32: Gestión de informe de incidentes

Tabla 39: Gestión de informe de incidentes

Nº	ENTRADA	ACTIVIDAD	SALIDA	DESCRIPCIÓN	RESPONSABLE
-	-	Inicio	Se genera la necesidad de generar el reporte de incidentes.	Se inicia el Proceso de Gestión de Informe de Incidentes	Tester
1.	Se genera la necesidad de generar el reporte de incidentes.	Obtener registro de pruebas	Informe de Pruebas	Se obtiene el registro de Pruebas ejecutado	Tester
2.	Informe de Pruebas	Obtener Plan de Pruebas	Plan de Pruebas	Se obtiene el Plan de Pruebas	Tester

				seleccionado para el proceso	
3.	Plan de Pruebas	Análisis preliminar de Incidentes	Informe de Incidentes Parcial	Se analiza los resultados obtenidos.	Tester
4.	Informe de Incidentes Parcial	¿Análisis Válido?	Informe de Incidentes incompleto	Se evalúan los incidentes encontrados	Tester
			Informe de Incidentes aceptado		
5.	Informe de Incidentes aceptado	Realizar Análisis Técnico	Informe de Incidentes actualizado	Se realiza el análisis técnico de los incidentes encontrados	Tester
6.	Informe de Incidentes actualizado	Realizar análisis de Factibilidad	Informe de Incidentes actualizado	Se realiza el análisis de factibilidad de los incidentes.	Tester
7.	Informe de Incidentes actualizado	Análisis Finalizado	Análisis incompleto	Se evalúa si el análisis finalizó y se obtuvo el resultado final	Tester
			Análisis completo		
8.	Informe de Incidentes actualizado	Generar Informe de Incidentes	Informe de Incidentes completado	Se genera el informe de Incidentes final a partir de lo antes analizado	Tester

9.	Informe de Incidentes completado	Registrar Cierre de Pruebas	Informe de Incidentes registrado	Se registra el cierre de pruebas finales	Tester
	Informe de Incidentes registrado	Fin	-	Finaliza el proceso de Generación de Informes de Incidentes	Tester

CAPITULO 6: RESULTADO DEL PROYECTO

En este capítulo se detallan los resultados obtenidos de la implementación y operación del Modelo de Certificación de Pruebas No Funcionales.

MODELO DE CERTIFICACIÓN DE PRUEBAS NO FUNCIONALES

Luego de implementar el modelo de Certificación en la empresa GMD, definimos los indicadores que nos apoyarían a medir si la ejecución de lo planificado fue el esperado. Para ellos mostraremos los resultados de cada fase del desarrollo del modelo.

Modelo de Certificación

Se estableció un nuevo flujo para el proceso de pruebas no funcionales basado en la ISO 29119 y otras buenas prácticas internacionales, asimismo usamos como el estándar requerimientos y pruebas no funcionales del ISQTB y el SEI, de donde seleccionamos los requerimientos no funcionales acorde a nuestra investigación científica. Este flujo se encuentra detallado en el punto **5.4 DISEÑO Y CARACTERIZACIÓN DEL MODELO DE CERTIFICACIÓN DE PRUEBAS NO FUNCIONALES**. A raíz de lo implementado, se procedió con la especificación de indicadores en el siguiente punto.

Especificación de indicadores

Para poder evaluar el desempeño del modelo de certificación a implementar en la organización, se han definido ciertos indicadores de carácter cuantitativo que permitirán analizar los resultados obtenidos.

Indicador 1: Porcentaje de utilización de documentos por proyecto previo a la fase de pruebas

Este indicador permitirá medir el porcentaje de documentos que se elaboran para poder solicitar el proceso de certificación de los proyectos entrantes. El propósito de este indicador es la de poder medir si se usa correctamente la documentación esencial de manera que se logre conocer el proyecto que pasará por el proceso de certificación. La fórmula para su cálculo es:

$$I1 = \frac{N^{\circ} \text{ Entregables entregados}}{N^{\circ} \text{ Entregables establecidos}} \times 100\%$$

Como se desea indicar el porcentaje de entregables utilizados, este se evalúa a partir del siguiente recuadro:

Tabla 40 Cuadro de Evaluación de Indicador 1

Descripción	Color
Mayor a 90 %	Muy Bueno
Entre 45% y 90%	Regular
Mayor a 90%	Malo

Indicador 2: Número de recursos asignados para la fase de pruebas

Este indicador permitirá medir el número de recursos utilizados durante la fase de certificación. El propósito de este indicador es la de medir si se emplea la cantidad de recursos esperados de manera que no exista sobrecarga de trabajo.

La fórmula para su cálculo es:

$$I2 = \text{Número de recursos asignados}$$

Como se desea indicar el número de recursos asignados, este se evalúa a partir del siguiente recuadro:

Tabla 41 Cuadro de Evaluación de Indicador 2

Descripción	Color
Sólo un recurso	Muy Bueno
De 2 a 3 Recursos	Regular
De 3 a más	Malo

Se resalta que, para este indicador, el semáforo utilizado es muy estricto por políticas organizacionales, ya que demandaría un alza en los costos.

Indicador 3: Porcentaje de reducción de tiempo en la fase de pruebas

Este indicador permitirá medir el porcentaje del tiempo empleado en la organización para la fase de pruebas. Su propósito es la de medir el tiempo ahorrado en la ejecución de pruebas, logrando contemplar la agilidad que ofrece el modelo de certificación. La fórmula para su cálculo es:

$$I3 = \frac{\text{Horas empleadas}}{\text{Horas definidas}} \times 100\%$$

Para la medición de este indicador, se tiene como base que para la planificación del proyecto que se le asigne el 30% del total del proyecto a las pruebas. Partiendo de lo antes mencionado, como se desea buscar el porcentaje de ahorro de tiempo, este se evalúa según el siguiente recuadro:

Tabla 42 Cuadro de Evaluación de Indicador 3

Descripción	Color
Menor o Igual a 30%	Muy Bueno
Entre 31% y 60%	Regular
Entre 61% y 100%	Malo

Indicador 4: Factor de Complejidad Ambiental

Este indicador, del cual GMD es el autor, ayuda a medir el índice de aceptación del modelo de certificación implementado. Su propósito es la de conocer el margen de aceptación que los colaboradores del área de testing. La fórmula para su cálculo es:

$$I4 = 1.4 + (-0.03 * \text{Promedio de Factores Ambientales})$$

Como se desea indicar el índice de Complejidad Ambiental, este se evalúa a partir del siguiente recuadro:

Tabla 43 Cuadro de Evaluación de Indicador 4

Descripción	Color
Menor a 0,90	Muy Bueno
Entre 0,91 y 1,30	Regular
Mayor a 1,31	Malo

Para halla el valor del Promedio de los Factores Ambientales, se parte de la evaluación realizada a un conjunto de colaboradores del área de testing de manera que se logre obtener su punto de vista frente a lo implementado. De esta manera se solicitó que califiquen bajo su criterio los siguientes puntos expuestos siguiente recuadro. Para mayor detalle consultar el **Anexo 7 – Matriz de Evaluación de Indicadores**.

Tabla 44 Cuadro de Cálculo de Factores Ambientales

Factores del ambiente y el equipo	Peso (P)	Valor Asignado (0-5) (VA)	Valor Promedio (VP)
Familiaridad con el modelo de pruebas utilizado	1.5	VA1 = Valor Promedio Asignado – Factor 1	$P * VA1$
Experiencia en la fase de pruebas	1	VA2 = Valor Promedio Asignado – Factor 2	$P * VA2$
Capacidad del Analista Líder	1	VA3 = Valor Promedio Asignado – Factor 3	$P * VA3$
Motivación	0.5	VA4 = Valor Promedio Asignado – Factor 4	$P * VA4$
Personal de medio tiempo	-1	VA5 = Valor Promedio Asignado – Factor 5	$P * VA5$
Dificultad en capacitación para	-1	VA6 = Valor Promedio Asignado – Factor 6	$P * VA6$

uso de herramientas			
Factores ambientales			Promedio de Factores Ambientales

Indicador 5: Porcentaje de Complejidad del Modelo de Certificación

Este indicador permitirá medir el porcentaje de complejidad del modelo de certificación a partir de lo calculado en el anterior indicador. Su propósito es la de medir el porcentaje de complejidad que tiene el área de testing frente a la implementación del modelo de certificación. Su fórmula de cálculo es:

$$I5 = \frac{\text{Factor de Complejidad Ambiental obtenido}}{\text{Factor de Complejidad Ambiental ideal}} \times 100\%$$

Como se desea indicar el porcentaje de complejidad del modelo de certificación, este se evalúa a partir del siguiente recuadro:

Tabla 45 Cuadro de Cálculo de Factores Ambientales

Descripción	Color
0%	Muy Bueno
Entre 1% y 33%	Bueno
Entre 34% y 66%	Regular
Entre 67% y 90%	Malo
Entre 91% y 100%	Muy Malo

Resultados de Evaluación del Modelo de Certificación

En esta sección se detallan los beneficios que trajo la implementación del modelo de certificación, expuestos a través de una matriz de evaluación detallada en el Anexo 7 – Matriz de Evaluación de Indicadores.

Esta evaluación realizada, gracias a los scores obtenidos, nos permite mostrar los resultados luego de la implementación del modelo, con la cual podremos interpretarlos y establecer acciones de mejora en la organización.

Para efectos de la validación del modelo, se nos fue otorgado 2 proyectos que contemplaban el despliegue de aplicaciones web, de manera que lográramos identificar cuáles eran esas brechas que no permitían asegurar la calidad de las pruebas no funcionales. De esa manera, posteriormente iniciamos con el flujo planteado contemplados en el inciso 5.4 – Diseño y caracterización del Modelo de Certificación, donde realizamos todas las actividades según lo definido. El detalle de las pruebas realizadas se referencia en el anexo correspondiente a las pruebas realizadas. Una vez finalizado el ciclo de pruebas, se procedió con la evaluación en la matriz que se muestra a continuación:

Tabla 46: Matriz de Resultados

Indicador	Objetivo	Fórmula	Unidad de Medida	Meta
Porcentaje de utilización de documentos por proyecto previo a la fase de pruebas	Determinar el porcentaje de documentos que se elaboran para poder solicitar el proceso de certificación de los proyectos entrantes	$(N^{\circ} \text{ entregables entregados} / N^{\circ} \text{ entregables establecidos}) * 100$	Porcentaje	100%
Número de recursos asignados para la fase de pruebas	Determinar el número de recursos utilizados durante la fase de certificación	Número de recursos asignados	Recurso	1
Porcentaje de reducción de tiempo en la fase de pruebas	Determinar el porcentaje del tiempo empleado en la organización para la fase de pruebas	Horas empleadas / Horas definidas	Horas	30%
Factor de Complejidad Ambiental	Determinar el factor ambiental que genera el modelo de certificación	$1.4 + (-0.03 * \text{Factores Ambientales})$	Índice de Complejidad Ambiental	0.8
Porcentaje de Complejidad del Modelo de Certificación	Determinar el porcentaje de Complejidad del modelo de certificación	$(\text{Factor de Complejidad ambiental obtenido} / \text{Factor de Complejidad ambiental obtenido}) * 100$	Porcentaje	100%

Los resultados que se muestran en la siguiente imagen, determinan el nivel de mejoría como parte de la adopción del modelo de certificación. Esto con el fin de mostrar que la implementación tuvo éxito, se plasmó también la evaluación de cada sistema proporcionado una vista clara de la situación actual por la que pasa la organización en su proceso de pruebas y que, gracias a la implementación del proyecto, dichas brechas encontradas sobre el conjunto de actividades anterior, ya fueron solucionadas.

Ilustración 33: Gráfica de presentación de Resultados

En cuanto al porcentaje de utilización de documentos del proyecto, existen proyectos que no generan documentación como el caso del primer sistema, ocasionando que se genere confusión en los testers ya que no hay una clara definición sobre qué pasará por el proceso de certificación.

Ilustración 34: Evaluación del Primer Sistema

Sistema: Sistema de Remuneraciones				Proceso de Pruebas Actual				Proceso de Pruebas / Modelo de Certificación			
Datos del Proyecto		Valorar	Unidad de Medida	Tiempo asignado Proyecto - meses				Tiempo asignado Proyecto - meses			
Cantidad de Horas por Recursos		140	Horas	Tiempo asignado Proyecto - Horas		960		Tiempo asignado Proyecto - Horas		480	
Porcentaje asignado para simulación de sistema		10%	Porcentaje	Tiempo de Desarrollo		971		Tiempo de Desarrollo		516	
Porcentaje asignado para periodo de pruebas		90%	Porcentaje	Tiempo para pruebas		188		Tiempo para pruebas		144	
				Recursos / Horas		320		Recursos / Horas		100	
				TOTAL HORAS PARA PROYECTO		960		TOTAL HORAS PARA PROYECTO		660	
Indicador	Objetivo	Norma	Unidad de Medida	Nota	Resultados obtenidos	Cantidad Ideal	Cantidad real	Resultados obtenidos	Cantidad Ideal	Cantidad real	
Porcentaje de utilización de documentos por proyecto previo a la fase de pruebas	Determinar el porcentaje de documentos que se elaboran para poder solicitar el proceso de certificación de los proyectos anteriores	(N° entregables entregados / N° entregables solicitados) * 100	Porcentaje	100%	0.00%	5	1	100.00%	5	5	
Número de recursos asignados para la fase de pruebas	Determinar el número de recursos utilizados durante la fase de certificación	Número de recursos asignados	Recursos	1	2	1	2	1	1	1	
Porcentaje de reducción de tiempo en la fase de pruebas	Determinar el porcentaje del tiempo empleado en la organización para la fase de pruebas	Horas empleadas / Horas definidas	Horas	50%	60.00%	144	188	25.80%	144	120	

Ilustración 35: Evaluación del Segundo Sistema

Sistema: Sistema de Créditos y Subvenciones				Proceso de Pruebas: Actual				Proceso de Pruebas: Modelo de Certificación			
Datos del Proyecto		Valores	Unidad de Medida	Tiempo asignado Proyecto: meses		Tiempo asignado Proyecto: horas		Tiempo asignado Proyecto: meses		Tiempo asignado Proyecto: horas	
Cantidad de Horas por Recurso		118.8	Horas	5		552.96		118.8		552.96	
Porcentaje asignado para construcción de sistema		10%	Porcentaje	123.03%		123.03%		123.03%		123.03%	
Porcentaje asignado para período de pruebas		10%	Porcentaje	159.888		159.888		95.88		95.88	
				118.8		118.8		118.8		118.8	
				552.96		552.96		TOTAL HORAS PARA PROYECTO		TOTAL HORAS PARA PROYECTO	
Indicador	Objetivo	Fórmula	Unidad de Medida	Meta	Resultados obtenidos	Cantidad ideal	Cantidad real	Resultados obtenidos	Cantidad ideal	Cantidad real	
Porcentaje de utilización de documentos por proyecto previo a la fase de pruebas	Determinar el porcentaje de documentos que se elaboraron para poder solicitar el proceso de certificación de los proyectos entrantes	$(N^{\circ} \text{ entregables entregados} / N^{\circ} \text{ entregables establecidos}) * 100$	Porcentaje	100%	100.00%	5	5	100.00%	5	5	
Número de recursos asignados para la fase de pruebas	Determinar el número de recursos utilizados durante la fase de certificación	Número de recursos asignados	Recurso	1	2	1	2	2	1	1	
Porcentaje de reducción de tiempo en la fase de pruebas	Determinar el porcentaje del tiempo empleado en la organización para la fase de pruebas	$\text{Horas asignadas} / \text{Horas definidas}$	Horas	80%	92.00%	95.88	100.000	92.00%	95.88	95.88	

Así como se realizó la evaluación de la documentación generada para el proceso de certificación, se revisó la utilización de recursos asignados, obteniendo una sobreasignación, superando los resultados esperados. Además, se contabilizó la cantidad de horas planteadas obteniendo una sobrecarga de recursos y también, nos percatamos que parte del tiempo utilizado para las pruebas, era utilizado para testear, de manera que no era eficiente el proceso que realizaban.

La evaluación de cada sistema emitió diferentes valores, para efecto de la matriz de evaluación del modelo, se sacó un promedio de los valores que permitían obtener una mejor obtención de los scores detallados en el documento QSCERT – Matriz de Evaluación de Indicadores.

Análisis de factibilidad económica

Análisis

Análisis Financiero Inicial

A continuación, se calculará el costo de pérdida económica en casos de que un aplicativo web o mobil tenga fallas en producción.

COSTO POR FALLAS EN AMBIENTES DE PRODUCCION		
Descripción	Cantidad	Base Legal
Penalizaciones por fallas	3 – 5 UITs	Contratos

Costo UIT(S/.)	4,050	D.S. N° 353-2016-EF
Costo Total Infracción (S/.)	12,150 – 20,250	

Tabla 47: Costos por fallas en ambiente de producción

Como se puede observar en la tabla 20, existe una pérdida económica considerable de existir caídas en el ambiente de producción de un aplicativo web o mobile. La cantidad de UIT es en base al OFICIO_2055_2015_MINEDU el cual se encuentra en <https://www.sunedu.gob.pe/reglamento-de-infracciones-y-sanciones-de-la-sunedu/> las infracciones varían entre 100-300 UIT.

En la tabla 21, se describen los gastos ocasionados por el riesgo de caídas en las aplicaciones web y mobiles desarrolladas por GMD.

PERDIDA ECONOMICA OCACIONADA POR RIESGO DE FALLAS EN LA APLICACIÓN	
Cantidad promedio de proyectos con pase a producción por mes	20
Cantidad promedio de proyectos con pase a producción por año	250
Pago de Penalidades en un proyecto (Soles)	12,150 - 20,250
Pago de penalidades en un mes (Soles)	243,000 - 506,250
Promedio de pago de penalidad por mes (Soles)	374,625
Pago de penalidades en un año (Soles)	2,916,000 – 6,075,000

Tabla 48: Pérdida Económica mensual

Costo de implementación

En esta sección se consideran todos los costos adheridos a la implementación del modelo de certificación de pruebas no funcionales.

Costo de Recursos Humanos

En la tabla 22, se puede ver el costo por hora de los roles involucrados en el diseño e implementación del modelo.

Costo de RRHH proyecto			
Rol	Abreviatura	Cantidad	Costo por hora (S/.)
Jefe de Proyecto	JP	2	S/. 25,00
QA	QA	1	S/. 18,00

Tabla 49: Costo de horas

En la tabla 23, se calcula el costo del proyecto teniendo 4 fases según lo indica el PMBOK.

Costo referencial del proyecto			
Fases	Abreviatura	Horas estimadas	Costo total (S/.)
Iniciación	JP	30	S/. 750,00
	QA	30	S/. 540,00
Planificación	JP	40	S/. 1.000,00
	QA	70	S/. 1.260,00
Ejecución	JP	50	S/. 1.250,00
	QA	100	S/. 1.800,00
Control y Cierre	JP	30	S/. 750,00
	QA	10	S/. 180,00

Subtotal		S/. 7.530,00	
<i>Costos inesperados</i>	<i>Porcentaje</i>	<i>Real</i>	
10 % adicional	10,00%	S/. 7.530,00	S/. 753,00
Total		S/. 8.283,00	

Tabla 50: Costo del Proyecto

Costo de Recursos de Tecnología

En la tabla 24, se detallan los costos de las herramientas para realizar las pruebas a aplicaciones web.

Herramientas	Dólares	Soles
Browser Stack	59	193,52
Dynatrace	180	590,4
IBM Rational AppScan	99	324,72
Rational Performance Tester	99	324,72
Total por mes	437,00	1433,36
Total por año	5244,00	17200,32

Tabla 51: Costo de Herramientas para pruebas a app web

En la tabla 25, se detallan los costos de las herramientas para realizar las pruebas a aplicaciones móviles.

Herramientas	Dólares	Soles
IBM Security Appscan	99	324,72
LoadComplete Mobile	395	1295,6
Perfecto Mobile	129	423,12

Scirocco Cloud	350	1148
Total por mes	973,00	3191,44
Total por año	11676,00	38297,28

Tabla 52: Costo de Herramientas para pruebas a app mobiles

Costo total del proyecto

Se realizan los costos totales del modelo:

Costo total del modelo de certificación	
Descripción	Total
Costo del proyecto	8283,00
Costo de las herramientas de testing para web (Anual)	17200,32
Costo de las herramientas de testing para mobile (Anual)	38297,28
Costo Total	63780,60

Tabla 53: Costo Total

Análisis Financiero Final

Se calculan los costos del proyecto tanto en recursos humanos como tecnológicos y se compararon con la posible infracción de no adoptar medidas para asegurar la calidad de las aplicaciones desarrolladas por GMD. Se muestra la viabilidad económica del modelo en la tabla 27:

Beneficio de Implementación del Modelo	
Penalidades por fallo de SW	S/. 374625,00

Costo Total de Implementación del Modelo	S/. 63780,60
Gasto que se Ahorraría GMD	S/. 310844,40

Tabla 54: Beneficio de implementación

Se puede concluir que la propuesta es viable ya que reduce los gastos que puede ocasionar el riesgo que las aplicaciones web y móviles diseñadas presenten fallas en el ambiente de producción.

CAPITULO 7: GESTIÓN DEL PROYECTO

En este capítulo se describen las actividades llevadas a cabo para la correcta gestión del proyecto tales como actividades, hitos, entregables, fechas, colaboradores, entre otros. Asimismo, se detallan las gestiones realizadas en relación a todo lo planificado del proyecto.

Producto final

Se implementará un modelo de certificación basados en técnicas de validación y verificación dinámica en el proceso de pruebas no funcionales pertenecientes a los proyectos que contemplen en su alcance la implementación de una aplicación web o móvil de la empresa GMD. Asimismo, se brindarán las políticas y estrategias organizacionales que la organización debe seguir para que la gestión de las pruebas se realice de manera eficiente. Además, se propondrá herramientas que no incurran en un costo en específico para que las pruebas no funcionales se realicen basadas en un estándar internacional. Finalmente, se contará también con un gestor de incidentes, con lo que los recursos disponibles para que se automatice el registro de los mismos.

El alcance del proyecto está conformado por los siguientes puntos:

- Identificar el estado actual de la empresa GMD sobre las pruebas No Funcionales de las aplicaciones Web y Móviles de tal manera que se propongan oportunidades de mejora.
- Realizar el análisis FODA de los procesos de certificación de las fábricas de testing.
- Implementación de un Modelo de Certificación de empresa GMD basándose en estándares, metodologías y buenas prácticas internacionales sobre las pruebas de validación y verificación dinámicas de los requerimientos no funcionales.
- Elaborar un listado con los indicadores para comprobar la calidad del Modelo de Certificación sobre las aplicaciones web y móviles de los diversos proyectos.
- Validar el Modelo de Certificación con proyectos que tengan en su alcance la implementación de aplicaciones web y/o móviles para comprobar calidad de la implementación.
- Implantar una herramienta para la gestión de incidentes basadas en las Pruebas No Funcionales para realizar el seguimiento de los errores encontrados durante la ejecución de las pruebas para requerimientos no funcionales.
- Proponer un Modelo de Certificación de técnicas de validación y verificación dinámica para la empresa virtual IT-Service.

Gestión del Tiempo

Para esta primera etapa del proyecto, se ha cumplido con todo lo estipulado en el cronograma de actividades, desarrollando las actividades plasmadas en él, consiguiendo finalizar los artefactos en los tiempos acordados.

		Código - (QSCERT)	
1.1		- Inicio	6 días
1.1.1		+ Semana 1	6 días
1.2		- Planificación	24 días
1.2.1		- Semana 2	7 días
1.2.1.1		- Reunión con el Profesor Gerente de IT Expert	1 día
1.2.1.1.1		Presentación de Project Charter	1 hr
1.2.1.2		+ Reunión con el Profesor Gerente de QS	0.13 días
1.2.1.3		+ Elaboración de Project Charter	1 día
1.2.1.4		+ Reunión con el Profesor Cliente	2 días
1.2.1.5		- Elaboración de Documentos de Gestión	2 días
1.2.1.5.1		Plan de Gestión de Alcance	4 hrs
1.2.1.5.2		Plan de Gestión de Calidad	3 hrs
1.2.1.5.3		Plan de Gestión de RRHH	5 hrs
1.2.1.6		+ Avance del Documento Memoria del Proyecto	1 día
1.2.2		- Semana 3	7 días
1.2.2.1		- Elaboración de Documentos de Gestión	5 días
1.2.2.1.1		Plan de Gestión de Comunicaciones	2 hrs
1.2.2.1.2		Registro de interesados	2 hrs
1.2.2.1.3		Plan de Gestión de Cronograma	3 hrs
1.2.2.1.4		Cronograma	5 hrs
1.2.2.1.5		Diccionario de EDT	8 hrs
1.2.2.1.6		Plan de Gestión de Riesgos	2 hrs
1.2.2.1.7		Matriz RAM	2 hrs
1.2.2.1.8		Matriz de Trazabilidad de Requerimientos	3 hrs
1.2.2.1.9		Matriz de Riesgos	4 hrs
1.2.2.1.10		Matriz de Comunicaciones	2 hrs
1.2.2.1.11		Descripción de Roles y Responsabilidades	4 hrs
1.2.2.2		+ Reunión con el Alumno Gerente de QS	1 día
1.2.2.3		+ Avance del Documento Memoria del Proyecto	4 días
1.2.2.4		+ Reunión con el Profesor Gerente de IT Expert	1 día
1.2.2.5		+ Reunión con el Profesor Cliente	1 día
1.2.2.6		- Elaboración de Documentos de Gestión	2 días
1.2.3		+ Semana 4	6 días
1.2.4		+ Semana 5	6 días
1.3		- Ejecución	58 días
1.3.1		+ Semana 6	7 días
1.3.2		+ Semana 7	6 días
1.3.3		+ Semana 8	6 días
1.3.4		- Semana 9	6 días

1.3.4.1		Elaboración del Modelo de Certificación	5 días
1.3.4.1.1		Benchmarking de pruebas no funcionales para QS	5 hrs
1.3.4.1.2		Benchmarking de herramientas de pruebas no	5 hrs
1.3.4.1.3		Propuestas de Políticas y Lineamientos de Calidad para QS	4 hrs
1.3.4.1.4		Diseño de la propuesta del proceso de certificación para QS	4 hrs
1.3.4.1.5		Plan de pruebas de V&V dinámica para requerimientos no funcionales para QS	5 hrs
1.3.4.2		Reunión con el Profesor Gerente de IT Expert	1 día
1.3.4.3		Elaboración de Modelo de certificación	1 día
1.3.4.4		Reunión con el Profesor Gerente de QS	1 día
1.3.4.5		Reunión con el Alumno Gerente de QS	1 día
1.3.4.6		Revisión de Documentos de Gestión por QS	1 día
1.3.4.7		Reunión con el Profesor Cliente	1 día
1.3.4.8		Elaboración de Modelo propuesto	2 días
1.3.4.9		Avance del Documento Memoria del Proyecto	1 día
1.3.4.9.1		Avance del cuarto capítulo de la Memoria	4 hrs
1.3.5		+ Semana 10	6 días
1.3.6		+ Semana 11	6 días
1.3.7		+ Semana 12	6 días
1.3.8		+ Semana 13	6 días
1.3.9		+ Semana 14	6 días
1.3.10		+ Semana 15	6 días
1.3.11		+ Semana 16	6 días
1.3.11.1		Hito 4: Aprobación del Modelo de Certificación Propuesto por parte del Comité de Proyectos	0 días
1.3.11.2		Segunda Sustentación del Proyecto	3 hrs
1.4		- Control	45 días
1.4.1		- Semana 17	6 días
1.4.2		- Semana 18	6 días
1.4.3		- Semana 19	6 días
1.4.4		- Semana 20	6 días
1.4.5		- Semana 21	6 días
1.4.5.1		Reunión con el Profesor Gerente de IT Expert	2 hrs
1.4.5.2		Reunión con el Profesor Cliente	2 hrs
1.4.5.3		Actividades de control	2 hrs
1.4.6		- Semana 22	6 días
1.4.6.1		Hito 5: Aprobación de control por parte del Comité de Proyectos	0 días
1.4.6.2		Tercera Sustentación del Proyecto	3 hrs
1.4.7		- Semana 23	6 días
1.4.8		- Semana 24	6 días
1.4.8.1		Reunión con el Profesor Gerente de IT Expert	2 hrs
1.4.8.2		Reunión con el Profesor Cliente	2 hrs
1.4.8.3		Actividades de control	5 hrs

1.4.9		- Semana 25	6 días
1.4.9.1		Reunión con el Profesor Gerente de IT Expert	2 hrs
1.4.9.2		= Reunión con el Profesor Cliente	1 día
1.4.9.2.1		Documento de aceptación de Proyecto	1 hr
1.4.9.2.2		Modelo de certificación validado	1 hr
1.4.9.2.3		Flujo de trabajo Final	1 hr
1.4.9.3		Actividades de control	4 hrs
1.5		- Cierre	36 días
1.5.1		+ Semana 26	6 días
1.5.2		+ Semana 27	6 días
1.5.3		+ Semana 28	6 días
1.5.4		+ Semana 29	6 días
1.5.5		+ Semana 30	6 días
1.5.6		- Semana 31	6 días
1.5.6.1		Reunión con el Profesor Gerente de IT Expert	2 días
1.5.6.2		= Reunión con el Profesor Cliente	1 día
1.5.6.2.1		Acta de Cierre	1 hr
1.5.6.2.2		Memoria Final	1 hr
1.5.6.2.3		Poster del Proyecto	1 hr
1.5.6.3		Actividades de control	3 hrs
1.5.7		- Semana 32	6 días
1.5.7.1		Hito 5: Aprobación de control por parte del Comité de Proyectos	0 días
1.5.7.2		Sustentación final del Proyecto	3 hrs

Ilustración 36: Cronograma de Proyecto

Ilustración 37: Árbol de entregables 1ra parte

Ilustración 38 Árbol de entregables 2da parte

Gestión de los Recursos Humanos

A continuación, se presentan los recursos que intervinieron a lo largo del desarrollo de la primera parte del proyecto.

Rol	Miembro	Descripción
Comité de Dirección	Rosario Villalta Jimmy Armas Luis Vives Pedro Shiguihara	Encargados de la evaluación y aprobación de los diversos proyectos que desarrollan las empresas virtuales de la UPC.
Gerente General de Empresas Virtuales	Emilio Herrera	Encargado de la gestión de la cartera de proyectos de la EISC, asimismo de la gestión de las empresas virtuales.
Gerente Profesor de DataCenter	Paul Rivas	Encargado de apoyar la gestión de los proyectos asignados a la empresa DataCenter y también del cumplimiento de los objetivos.
Jefes de Proyecto	Manuel Quezada Renzo Villarreal	Encargados de la gestión y desarrollo del proyecto asignado.
Recurso QS	Asignado IT-Services	Encargado de validar los artefactos enviados por los Jefes de Proyectos, además de apoyar en las pruebas de validación del proyecto.

Tabla 55: Tabla de RRHH

Gestión de las Comunicaciones

Las comunicaciones que se realizaban eran de la siguiente manera:

Recurso de Taller de Desempeño Profesional

La comunicación que se realizó con el recurso asignado a la revisión de los artefactos a lo largo de la primera etapa fue presencial, se citaba en el aula de DataCenter como el de IT-Services para coordinar y aclarar las dudas con respecto a las revisiones de los artefactos. Asimismo, se hacía uso del correo electrónico para enviar los artefactos e iniciar las revisiones y se reciben los informes post-revisión.

Ilustración 39: Correo de IT-Services

De la misma manera solicitamos un recurso para nos ayude a realizar las pruebas respectivas al modelo de certificación haciendo uso de las herramientas propuestas. Estas ya estaban pre configuradas en un ordenador capacitado de manera que se logre obtener los resultados esperados.

Profesor Cliente

Las reuniones con el Profesor cliente se realizaban presencialmente en la universidad para que nos de feedback de los avances, ahí mismo se desarrollaba y firmaba el acta, una vez firmada el acta se escaneaba y subía al drive. Asimismo, se realizaban call vía Skype para aclarar dudas frente al desarrollo de algún tema en específico.

Ilustración 40: Carpetas de Google Drive

Gerente Profesor y Alumno

La comunicación con el profesor y alumno gerente se realizaba en el horario de clases, si algún miembro del equipo tenía alguna duda, indicaba el profesor para poder reunirse y aclarar dudas con respecto al desarrollo del proyecto, luego se generaba el acta de reunión,

y, con respecto al alumno gerente, se consultaba el cronograma de actividades del ciclo para definir fechas de entregas de lo requerido.

Ilustración 41: Carpetas de Google Drive

Gestión de los Riesgos

Los riesgos que hemos encontrado sobre el desarrollo del proyecto se explican a continuación, asimismo, se explica la probabilidad de ocurrencia y la estrategia que se debe desarrollar para mitigarlos.

#	Riesgo	Probabilidad	Impacto	Estrategia de mitigación
1	Cancelación de las reuniones con el cliente o profesor gerente	Bajo	Media	Se pactarán reuniones por internet o durante la semana para subsanar este riesgo.
2	Cambio del alcance del proyecto por necesidades del cliente	Alto	Muy alta	Se realizará un ajuste al proyecto, considerando siempre el impacto en tiempo que este puede tener en los demás temas considerados dentro del alcance.

3	Incumplimiento de los entregables por parte del equipo	Medio	Muy alta	Se dedicará tiempos extras para cerrar los documentos pendientes.
4	No incurrir con el costo de las licencias para las herramientas a proponer	Medio	Alta	Se buscarán herramientas open source para que sean implementadas a la empresa virtual QS.
5	Disconformidad en la entrega de los informes ya desarrollados	Bajo	Muy alta	Se elaborará nuevamente el artefacto desarrollado, considerando las observaciones del Stakeholder involucrado.
6	La estimación del tiempo del proyecto es optimista, en vez de ser realista	Medio	Alta	Se debe elaborar el cronograma con tiempos realistas.
7	Incumplimiento de la corrección de los documentos a cargo del recurso de QS.	Medio	Media	Se debe realizar el seguimiento de la solicitud de corrección enviada a través de correo electrónico.

Tabla 56: Tabla de Riesgos

Lecciones aprendidas

- Frente a los diversos cambios que se realizan durante el desarrollo del proyecto, en las reuniones tanto con el cliente como con el profesor gerente, se deben generar las actas de reuniones de tal manera que se guarde la evidencia de lo conversado.
- Para poder proponer o implementar algún estándar que sea aplicable en una organización, se deben identificar las brechas entre la situación actual y el caso ideal, asimismo, conocer muy bien los procesos core del negocio de tal manera que se cubra lo necesitado.
- Para llegar a cumplir las fechas estipuladas en el cronograma final del proyecto, se debe avanzar con los artefactos y/o actividades en la medida de lo posible sin llegar al límite de fechas para que se alcancen los objetivos, esto conlleva a que se obtenga tiempo adicional para el avance de otras actividades pendientes para evitar retrasos en los desarrollos.

Conclusiones

- Producto de la implementación del Modelo de Certificación de Pruebas no Funcionales en GMD, se obtienen las siguientes conclusiones:
 - Se verifica que los proyectos puestos en marcha en el ambiente de producción, exponen mayor robustez en su capacidad de operativa. Gracias a que los recursos (testers) lograron adaptarse a la metodología, se obtuvo una mejora de 41,25% de factor ambiental.
 - Asimismo, se verifica que en la fase de Generación del Plan de Pruebas, los testers cumplen en entregar la documentación requerida como parte de la metodología de trabajo. Esto se ve reflejada en la mejora de un 50% en la entrega de documentación.
 - Además, se verifica que existe una optimización en las tareas de los testers, logrando obtener la reducción de 200% a 100%, lo cual refiere a que el Tester solo está asignado a un proyecto en su totalidad, y ya no a hacer tareas a la vez como en un inicio ocurría.
 - De la misma manera, la fase de pruebas expuso una mejora en los tiempos logrando reducirlos. En GMD, la gestión de proyectos define que sólo el 30% del proyecto se dedica a la fase de pruebas, lo cual con el conjunto de actividades inicial no se conseguía. Para un proyecto puesto a prueba con la metodología, se redujo de 60% a 21.88%, consiguiendo que la cantidad de horas del Tester disminuya, esto gracias a que ya se tenía mayor conocimiento del alcance de las pruebas a ejecutar.
- Sin embargo, aún queda una brecha que cubrir en cuanto al factor ambiental, que según la metodología establecida, refiere que el valor ideal para este indicador es del 100%. Producto de las pruebas realizadas (2 proyectos otorgados por GMD), se obtuvo un valor promedio de 128.13% lo cual representa un logro ya que se redujo 41.25% según lo expuesto en el cuadro anterior.

Recomendaciones

- Se recomienda que la empresa capacite a todo el personal que pertenezca al ambiente de pruebas con el uso de la metodología implementada para mantener la calidad de las pruebas durante la fase de pruebas.
- Se recomienda que la empresa GMD invierta en la implementación de herramientas que ayuden a automatizar el proceso de pruebas, de manera que se continúe con la reducción de tiempos.

Glosario

- Pruebas dinámicas: proceso de corrección o validación de un producto software en un ambiente de pruebas o productivo.
- Pruebas estáticas: proceso de validación de los documentos, artefactos o entregables generados dentro del ciclo de vida de un producto software.
- Benchmarking: proceso que implica la comparación de productos, servicios o procesos.
- Verificación y validación dinámica: proceso de testeo donde se mide y evalúa la calidad de un producto software.
- Empresa virtual: organización que solo funciona dentro de la EISC de la UPC. Desarrolla proyectos o brinda servicios.
- Memoria: Recopilación de toda la investigación. Contempla las motivaciones, los condicionantes del proyecto, la tecnología del proyecto, las normas para su explotación, el presupuesto y la evaluación del proyecto.
- PMBOK: Metodología que brinda y describe estándares, pautas y normas para la gestión de proyectos
- ISTQB: Metodología de testeo de software.
- ISO: Organización que genera estándares internacionales para la mejora de procesos, productos, etc.
- CMMI: Proceso para la medición y mejora de desarrollo, gestión o adquisición de un software.

Siglarlo

- **P.M.B.O.K.** Acrónimo de Project Management Body of knowledge (Guía de los Fundamentos para la Dirección de Proyectos)
- **ISTQB** International Software Testing Qualifications Board
- **SWEBOK** The Guide to the Software Engineering Body of Knowledge
- **ISO** International Organization for Standardization

Bibliografía

Referencias bibliográficas

- Denni Aldi Ramadhani, Siti Rochimah, Umi Laili Yuhana. (diciembre de 2015). Classification of Non-Functional Requirements Using Semantic-FSKNN Based ISO/IEC 9126. TELKOMNIKA, 13, 1456-1465.
- Amira A. Alshazly, Ahmed M. Elfatraty, Mohamed S. Abougabal. (28 June 2014). Detecting defects in software requirements specification. Alexandria Engineering Journal, 53, 513 - 527.
- Lan Lina, Jia Hea , Yong Zhanga , Fengguang Songb. (2015). ELSEVIER. Quality Assurance through Rigorous Software Specification and Testing: A Case Study, 62, 257 – 265.
- Mary-Luz Sánchez-Gordóna, * Lourdes Morenoa. (2014). Toward an integration of Web accessibility into testing processes. ELSEVIER, 27, 281 – 291.
- C.J. Torrecilla-Salinas a,* , J. Sedeñoa,b , M.J. Escalonaa , M. Mejías. (2016). Agile, Web Engineering and Capability Maturity Model Integration: A systematic literature review.. ELSEVIER, 71, 92 - 107
- International Software Testing Qualifications Board. (2011). Certified Tester Foundation Level Syllabus. ISTQB.
- Bourque, P., Fairley, R.. (2014). Guide to the Software Engineering Body of Knowledge. IEEE: IEEE Computer Society.
- Equipo del Producto CMMI. (2010). CMMI para Desarrollo. Software Engineering Institute: SEI.
- Fraunhofer FOKUS, Competence Center for Testing, Interoperability and Performance, The UML 2.0 Testing Profile and its Relation to TTCN-3, Competence Center for Testing, Interoperability and Performance.

Denni Aldi Ramadhani, Siti Rochimah, Umi Laili Yuhana. (2015). Classification of Non-Functional Requirements Using Semantic-FSKNN Based ISO/IEC 9126. TELKOMNIKA, Vol 13, 1456-1465.

Lu, L., Hong, Y., Huang, Y., Su, K., Yan, Y, Activity page based functional test automation for android applications, 2012

Nagowah, L., Sowamber, G., A novel approach of automation testing on mobile devices, 2012.

Song, H., Ryoo, S., Kim, J.H., An integrated test automation framework for testing on heterogeneous mobile platforms, Software and Network Engineerin, 2011.

Amalfitano, D., Fasolino, A.R., Tramontana, P., A GUI Crawling-Based Technique for Android Mobile Application Testing, 2011.

Villanes, I.K., Bezerra Costa, E.A., Dias-Neto, A.C Automated Mobile Testing as a Service 2015.

Lettner, F., Holzmann, C., Automated and unsupervised user interaction logging as basis for usability evaluation of mobile applications, 2012.

Adamsen, C.Q., Mezzetti, G., Møller, A., Systematic execution of Android test suites in adverse conditions. 2015.

Hay, R., Tripp, O., Pistoia, M., Dynamic detection of inter-application communication vulnerabilities in Android. 2015

Vieira, V., Holl, K., Hassel, M., A context simulator as testing for application mobile 2015.

<https://www.ieee.org/index.html>, 2015

<http://www.dynatrace.com/es/index.html>, 2014

Referencias electrónicas.

Working Group 26. (2014). The International Software Testing Standard. 2014, de ISO/IEC JTC1/SC7 WG26 Sitio web: <http://www.softwaretestingstandard.org/>

Paul Baker, Zhen Ru Dai, Jens Grabowski, Øystein Haugen, Serge Lucio, Eric Samuelsson, Ina Schieferdecker, and Clay E. Williams⁸. (2012). The UML 2.0 Testing Profile. 2012, de UML 2.0 Sitio web: <http://folk.uio.no/oystein/CONQUEST-FINAL>

Software Engineering Institute. (2016). Digital Library. 2016, de 2016 Sitio web: <https://resources.sei.cmu.edu/library/>

Anexos

Anexo 1 – Análisis FODA – Fábricas de Testing y Software

Objetivo

Encontrar modelos del proceso de pruebas funcionales y no funcionales para tomar de referencia en la elaboración del Modelo de Certificación de Pruebas No Funcionales.

Descripción

Se visitó 2 empresas que se especializan en el desarrollo y testing de software, son conocidas como fábricas de Testing y Fábricas de Software, las cuales obtuve de referencia por compañeros de trabajo. Al ir a cada Fábrica, la entrevista que realicé se basó en conocer el proceso principal de pruebas funcionales y no funcionales de las aplicaciones web y móviles, asimismo, entrevisté al líder de QA, quien no quiso brindarme mucha información debido a la confidencialidad que guarda la organización, pero en la deducción logré identificar los principales criterios para realizar el análisis FODA de cada uno, de tal manera que pueda desarrollar un modelo de certificación para pruebas No Funcionales.

En tal sentido, se desarrolló los análisis a continuación, detallando cada punto tocado en las entrevistas. Cabe resaltar que no se obtuvo mucha información, y que por referencia de algunos estándares y metodologías, logré identificar también algunos de los puntos contemplados en este análisis. Y también, solicitaron que se guarde la confidencialidad en cuanto al nombre de las organizaciones, de tal manera que no se vea afectada la imagen.

Análisis FODA

ANÁLISIS FODA DE EMPRESA “Anónimo1”

Esta empresa que fue entrevistada, nos brindó información no tan detallada con respecto a sus pruebas no funcionales, lo poco que nos mostraron nos ayudó poco para poder realizar el Análisis FODA, es por ello que, en la información, mencionó que usaba ciertos estándares

y logramos deducir ciertos aspectos que serán expuestos a continuación. Asimismo, no poseen un marco de referencia sino usan una metodología elaborada por la propia organización basado en un estándar de calidad, pero no basta con ello para poder certificar que las aplicaciones que salen a producción.

FORTALEZAS	OPORTUNIDADES
<p>Poseen la infraestructura adecuada para poder realizar las pruebas no funcionales.</p> <p>Contempla diversos estándares y buenas prácticas en cuanto a la operatividad y soporte de Infraestructura.</p> <p>Productos bien diferenciados y organizados.</p> <p>Política de Sistemas e infraestructura adecuada para las aplicaciones y servicios que brindan.</p> <p>Buen ambiente de trabajo</p>	<p>Existen varios estudios para proponer un modelo que permita brindar mayor calidad a las operaciones del negocio.</p> <p>Existe el incentivo de aprovechar diferentes responsabilidades de otros usuarios para apoyar los procesos de pruebas.</p> <p>Competencias débiles.</p> <p>Crecimiento más rápido en el mercado de pruebas</p>
DEBILIDADES	AMENAZAS
<p>Tiempos de respuesta elevados</p> <p>Proceso de Pruebas sin estándar</p> <p>Falta de capacitación.</p>	<p>Conflictos sobre el aseguramiento de la calidad</p> <p>Cambios constantes de responsabilidades y recursos por falta de capacitación.</p>

<p>No existe una dirección estratégica clara.</p> <p>Falta de Habilidades o capacidades.</p> <p>Recursos financieros insuficientes</p>	<p>Los competidores por a veces las fallas buscan otro proveedor de Pruebas y desarrollo.</p>
--	---

ANÁLISIS FODA DE EMPRESA “Anónimo2”

Esta empresa se encontró más alineada con un estándar de pruebas, trayendo consigo una serie de oportunidades de mejoras que se contemplan a continuación. Parte de la entrevista se enfocó en lo relacionado a la infraestructura que posee que soporte los requerimientos no funcionales, con lo que nos comentaron que su debilidad es el soporte técnico en cuanto a requerimientos no funcionales.

FORTALEZAS	OPORTUNIDADES
<p>Tiene una línea de productos y herramientas para las pruebas funcionales y no funcionales.</p> <p>Posee estándares de calidad para dar mejores productos a los clientes</p> <p>Cualidades del servicio son considerados de alto nivel según el mercado.</p>	<p>El mercado cada vez crece más y los productos deben tener la calidad esperada.</p> <p>El estado ofrece más productos cada día y necesita proveedores confiables.</p> <p>La poca variedad de los cursos que dicta en cuanto a desarrollo y estandarización se refiere.</p>
DEBILIDADES	AMENAZAS

El capital del trabajo es mal utilizado, no invierten en infraestructura.	La poca identificación de la marca, al no ser conocida, sus ingresos son bajos.
No tiene una alineación estratégica para el desarrollo y pruebas de los sistemas y aplicaciones.	Al ser bajos los recursos no pueden obtener nuevas tecnologías.
Su modelo de Referencia es obsoleto.	Los mejores recursos de la organización pueden retirarse para encontrar una nueva oportunidad.
Los tiempos de respuesta se atrasan.	

Anexo 2 – Política Organizacional de Pruebas

Declaración de Políticas de Pruebas

Los Testers de la empresa GMD son los únicos responsables de utilizar las técnicas de validación y verificación estática y dinámica del proceso de pruebas funcionales y no funcionales de aquellos proyectos web y móviles. Asimismo, debe velar por la capacitación y asignación de las pruebas realizadas al jefe de Testing de manera que esté enterado de lo ejecutado.

Objetivos Organizacionales de Pruebas

El presente documento tiene por objetivo establecer los principales lineamientos, actividades y responsabilidades para la gestión de requerimientos no funcionales para el proceso de pruebas de las aplicaciones web y móviles. Asimismo, su objetivo también es reducir el impacto sobre las mejoras de las aplicaciones.

Alcance

Todo proceso de pruebas que considere un programa, procedimiento o estructura de pruebas debe ajustarse a todas las consideraciones establecidas en este documento. Se debe tener en claro que, bajo los conceptos explicados, las modificaciones que se realicen previamente a las ejecuciones de las pruebas no solo abarcan a los testers, sino también a los encargados de mantener la infraestructura. Este documento debe ser de conocimiento del personal de pruebas.

Documentos Relacionados

QSCERT - Definiciones y Terminología del Modelo de Certificación

Políticas

- Las herramientas oficiales con la que se realizarán las pruebas de aplicaciones web son:
 - Load Impact
 - Dynatrace
 - Browser Stack
 - Selenium WebDriver
 - Rational Performance Tester
 - Soasta CloudTest
 - IBM Rational AppScan
 - OWASP Zap
- Las herramientas oficiales con la que se realizarán las pruebas a aplicaciones móviles son:
 - Perfecto Mobile
 - User Testing
 - LoadComplete Mobile
 - JMeter

- IBM Security AppScan
- Scirocco Cloud
- El Modelo describe que las pruebas pasarán por las 3 capas definidas en el Modelo de Certificación elaborado, los cuales son:
 - Capa Organizacional de Pruebas
 - Capa de Gestión de Pruebas
 - Capa de Gestión de Pruebas Dinámicas
- En general, no se admitirán modificaciones mientras estén en ejecución las pruebas correspondientes, salvo la naturaleza de la falla o problema sea en la infraestructura del ambiente, el cambio se realizará con la autorización del Líder o encargado de Infraestructura.
- En el Procedimiento de Pruebas debe contemplar los pasos para poder retornar al estado inicial, previo al inicio de las pruebas.
- El Jefe de Proyecto debe ser responsable de monitorear los cambios a realizar en los ambientes de pruebas para que la ejecución sea continua.
- El encargado de las pruebas debe estar altamente capacitado frente a cualquier incidente que se presente durante las pruebas.
- Toda solicitud de Pruebas será rechazado y notificado al Jefe de Proyecto si no contiene como mínimo la siguiente información:
 - Solicitud de Pruebas o Especificación de requerimiento del aplicativo web o móvil
 - Código Fuente del requerimiento
- Se entregará el Certificado de Calidad cuando las pruebas hayan obtenido la aprobación de parte de los Gestores de las pruebas, en este caso el Jefe de Testing.

Código de Ética

Los colaboradores del área de testing de la empresa GMD, deberán estar capacitados y ser expertos en pruebas, asimismo deben estar sujetos a los principios del código de ética que se establecen a continuación:

- **Sociedad:** Los recursos actuarán en forma congruente con el interés social.
- **Cliente:** Los recursos actuarán de manera que se concilien los mejores intereses con los Jefes de Proyectos.
- **Producto:** Los recursos asegurarán que sus pruebas e informes cumplen con estándares planteados por la empresa GMD
- **Juicio:** Los recursos mantendrán la integridad e independencia en su juicio profesional.
- **Administración:** El Profesor y alumno gerente de la empresa virtual QS deberán mantener el enfoque ético sobre la administración de las pruebas de aplicaciones web y móviles.
- **Profesión:** Los recursos incrementarán la integridad y reputación sobre las pruebas a realizar.
- **Colegas:** Los recursos apoyarán y serán justos con sus colegas.
- **Personal:** Los recursos participarán en el aprendizaje relacionado a las prácticas de validación y verificación estática y dinámica de la empresa GMD

Excepciones

Cualquier excepción a las políticas deberá ser autorizada por el jefe o encargado de Infraestructura.