



Autónoma
Universidad Autónoma del Perú

FACULTAD DE INGENIERÍA Y ARQUITECTURA
ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS

TRABAJO DE SUFICIENCIA PROFESIONAL
PROCESO DE CALIDAD DE SOFTWARE EN EL PROCESO DE RESTABLECER
CLAVE CON BIOMETRÍA FACIAL EN EL APLICATIVO DE TUNKI

PARA OBTENER EL TÍTULO DE
INGENIERA DE SISTEMAS

AUTORA

ANA MARIBEL SARAVIA HUAMANI
ORCID: 0000-0002-6376-5145

ASESOR

DR. JULIO ELVIS VALERO CAJAHUANCA
ORCID: 0000-0002-8522-6249

LÍNEA DE INVESTIGACIÓN

DESARROLLO DE SOFTWARE

LIMA, PERÚ, ABRIL DE 2023



CC BY

<https://creativecommons.org/licenses/by/4.0/>

Esta licencia permite a otros distribuir, mezclar, ajustar y construir a partir de su obra, incluso con fines comerciales, siempre que le sea reconocida la autoría de la creación original. Esta es la licencia más servicial de las ofrecidas. Recomendada para una máxima difusión y utilización de los materiales sujetos a la licencia.

Referencia bibliográfica

Saravia Huamani, A. M. (2023). *Proceso de calidad de software en el proceso de restablecer clave con biometría facial en el aplicativo de Tunki* [Trabajo de suficiencia profesional, Universidad Autónoma del Perú]. Repositorio de la Universidad Autónoma del Perú.

HOJA DE METADATOS

Datos del autor	
Nombres y apellidos	Ana Maribel Saravia Huamani
Tipo de documento de identidad	DNI
Número de documento de identidad	47645533
URL de ORCID	https://orcid.org/0000-0002-6376-5145
Datos del asesor	
Nombres y apellidos	Julio Elvis Valero Cajahuanca
Tipo de documento de identidad	DNI
Número de documento de identidad	80543932
URL de ORCID	https://orcid.org/0000-0002-8522-6249
Datos del jurado	
Presidente del jurado	
Nombres y apellidos	Carlos Alberto Lon Kan Prado
Tipo de documento	DNI
Número de documento de identidad	15595507
Secretario del jurado	
Nombres y apellidos	Ivonne Sadith Musayon Oblitas
Tipo de documento	DNI
Número de documento de identidad	09606289
Vocal del jurado	
Nombres y apellidos	Julio Elvis Valero Cajahuanca
Tipo de documento	DNI
Número de documento de identidad	80543932
Datos de la investigación	
Título de la investigación	Proceso de calidad de software en el proceso de restablecer clave con biometría facial en el aplicativo de Tunki
Línea de investigación Institucional	Ciencia, Tecnología e Innovación
Línea de investigación del Programa	Desarrollo de Software
URL de disciplinas OCDE	https://purl.org/pe-repo/ocde/ford#2.02.04



**FACULTAD DE INGENIERÍA Y ARQUITECTURA
ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS**

ACTA DE SUSTENTACIÓN DE TRABAJO DE SUFICIENCIA PROFESIONAL

El jurado evaluador del Trabajo de Suficiencia Profesional:

**“PROCESO DE CALIDAD DE SOFTWARE EN EL PROCESO DE RESTABLECER CLAVE
CON BIOMETRÍA FACIAL EN EL APLICATIVO DE TUNKI.”**

Que ha (n) sustentado:

SARAVIA HUAMANI, ANA MARIBEL

Apellidos

Nombre(s)

INTERESADO (DA) en optar el Título Profesional de:

INGENIERO DE SISTEMAS

ACUERDA APROBAR POR UNANIMIDAD

Lima, 13 de Abril del 2023

Presidente (a) de Jurado: MG. CARLOS ALBERTO LON KAN PRADO

Nombre Completo

Firma

Miembro (a) de Jurado: DRA. IVONNE SADITH MUSAYON OBLITAS

Nombre Completo

Firma

Miembro (a) de Jurado: DR. JULIO ELVIS VALERO CAJAHUANCA

Nombre Completo

Firma

**Decano de la Facultad de
Ingeniería y Arquitectura**

ACTA DE APROBACIÓN DE ORIGINALIDAD

Yo, Julio Elvis Valero Cajahuanca docente de la Facultad de Ingeniería y Arquitectura y Escuela Profesional de Ingeniería de Sistemas de la Universidad Autónoma del Perú, en mi condición de asesor del Trabajo de Suficiencia Profesional titulada:

PROCESO DE CALIDAD DE SOFTWARE EN EL PROCESO DE RESTABLECER CLAVE CON BIOMETRÍA FACIAL EN EL APLICATIVO DE TUNKI

del Bachiller (es):

ANA MARIBEL SARAIVIA HUAMANI

Constato que el informe tiene un índice de similitud de 7% verificable en el reporte de originalidad del programa Turnitin que se adjunta.

El analizó dicho reporte y concluyó que cada una de las coincidencias detectadas no constituyen plagio. A mi leal saber y entender la tesis cumple con todas las normas para el uso de citas y referencias establecidas por la Universidad Autónoma del Perú.

Lima, 12 de mayo del 2023



VALERO CAJAHUANCA JULIO ELVIS
Asesor
DNI: 80543932

DEDICATORIA

Este informe va dirigido hacia mis abuelos Pedro y Gertrudis quienes me han apoyado en todo momento e impulsando a seguir adelante ya que nunca es tarde para finalizar el ciclo de la universidad.

AGRADECIMIENTOS

Agradecer a la a mi casa de estudios Universidad Autónoma del Perú por darme la oportunidad de desarrollarme como estudiante dentro de su establecimiento y brindarme los conocimientos requeridos y necesarios en mi formación como futura profesional. A mi asesor es su momento Dr. Julio Elvis Valero Cahahuanca por sus acertadas sugerencias, correcciones y la motivación que me brindaba para continuar con mi objetivo. Un agradecimiento muy especial a mi familia que son el soporte de todo y me ha motivado para seguir adelante con mi titulación, del mismo modo a un gran amigo Bruno Saravia a mi otro estimado amigo Chito Saravia por su apoyo moral e incondicional.

ÍNDICE

DEDICATORIA	2
AGRADECIMIENTOS	3
RESUMEN	8
INTRODUCCIÓN	9
CAPÍTULO I: ASPECTOS GENERALES DEL TEMA	
1.1. Antecedentes	10
1.2. Descripción organizacional	11
1.3. Contexto socioeconómico	11
1.3.1. Oportunidades	11
1.3.2. Amenazas	12
1.4. Descripción general de experiencia	12
1.5. Explicación del cargo, funciones ejecutadas	12
1.6. Propósito del puesto	13
1.7. Producto o proceso	13
1.8. Resultados	14
CAPÍTULO II: FUNDAMENTACIÓN	
2.1. Aplicaciones teóricas	16
2.1.1. Pruebas del software	16
2.1.2. Principios generales de las pruebas.....	17
2.1.3. Técnicas de pruebas	20
2.1.4. Pruebas dinámicas.....	20
2.1.5. Pruebas funcionales.....	22
2.1.6. Pruebas no funcionales.....	24
2.1.7. Pruebas basadas en las experiencias	25
2.1.8. Pruebas estáticas	25
2.2. Métodos y procedimientos	26
2.2.1. Norma ISO/IEC 9126 : 2001	26
2.2.2. Proceso de pruebas del ISTQB.....	26
CAPÍTULO III: APORTES Y DESARROLLO DE EXPERIENCIAS	
3.1. Aportes teóricos	29
3.1.1. Objetivo general.....	29
3.1.2. Realidad problemática	29

3.2. Aportes prácticos	30
3.2.1. Refinamiento.....	30
3.2.2. Planificación y control	32
3.2.3. Análisis y diseño - sprint 1.....	34
3.2.4. Implementación y ejecución	36
3.2.5. Evaluación de los criterios de salida, creación de informes y cierre	38

CONCLUSIONES

RECOMENDACIONES

REFERENCIAS

ANEXOS

LISTA DE TABLAS

Tabla 1	Tipo de pruebas no funcionales
Tabla 2	Elaboración del example mapping con respecto a la regla 1 y su respectivo ejemplo con respecto a la historia de usuario
Tabla 3	Elaboración del example mapping con respecto a la regla 2 y su respectivo ejemplo con respecto a la historia de usuario
Tabla 4	Elaboración del example mapping con respecto a la regla 3 y su respectivo ejemplo con respecto a la historia de usuario
Tabla 5	Aspectos generales del plan de pruebas de historia de usuario
Tabla 6	Estrategia de pruebas del plan de pruebas
Tabla 7	Tipos de pruebas del plan de pruebas
Tabla 8	Información del diseño de caso de prueba
Tabla 9	Información de los pasos del caso de prueba 02
Tabla 10	Estados de la ejecución de los casos de prueba
Tabla 11	Información que debe contemplar un defecto
Tabla 12	Resultado de pruebas no funcionales
Tabla 13	Defectos encontrados en las pruebas

LISTA DE FIGURAS

- Figura 1 Representación del costo de corregir un bug en el ambiente de producción
- Figura 2 Representación gráfica de la agrupación de defectos por cantidad y por funcionalidad
- Figura 3 Interpretación gráfica de las pruebas de caja negra
- Figura 4 Interpretación gráfica de las pruebas de caja blanca
- Figura 5 Interpretación las pruebas de integración
- Figura 6 Definición de las pruebas de aceptación junto con el usuario que realizará las pruebas a la aplicación
- Figura 7 El proceso de pruebas de ISTQB y la cronología del proyecto
- Figura 8 Modelo del formato para diseñar los casos a ejecutar
- Figura 9 Casos de prueba importado en la herramienta Testlink
- Figura 10 Casos de prueba ejecutados en la herramienta Testlink
- Figura 11 Estado de los casos de prueba
- Figura 12 Gráfica de la evidencia adjunta en Testlink
- Figura 13 Resumen de ejecución de casos de prueba
- Figura 14 Estadísticas del esfuerzo de certificación
- Figura 15 Valoración de las pruebas

RESUMEN

El presente trabajo de suficiencia profesional tuvo como objetivo implementar un factor de seguridad en la funcionalidad de reestablecer contraseña, el cual consistió en lo siguiente: Cuando un usuario tenga la necesidad de cambiar su contraseña ya sea porque no la recuerda o simplemente requiere cambiarla entonces al iniciar el flujo ingresará su número de documento de identidad nacional, posterior a esto le llegará una clave SMS el cual se tendrá que ingresar para validar que le pertenece el celular al usuario, luego de ello iniciará por un flujo de biometría facial en el cual deberá tomarse una selfie para validar que sea la misma persona que se registró en el aplicativo TUNKI, caso contrario si no se logra validar la identidad del usuario entonces no logrará cambiar a una nueva contraseña. Para la implementación de esta nueva funcionalidad se realizó el proceso de calidad de software en todo el ciclo de desarrollo para garantizar el correcto funcionamiento del producto, planteando así estrategias de pruebas y determinados tipos de pruebas a realizar. Como resultado de este informe se logró que los usuarios tengan una mejor experiencia al realizar un cambio de contraseña, esto de una manera segura ya que al solicitarle el selfie para validar su identidad su identidad se está evitando y mitigando los casos de fraudes el cual puede repercutir en malestares para el usuario.

INTRODUCCIÓN

En el presente trabajo de suficiencia profesional fue desarrollado en base a mi experiencia y el marco de labores que realicé como Ingeniera de pruebas de software, a continuación se detallarán las labores desempeñadas y funciones en los próximos capítulos: Iniciemos con el Capítulo I, se da a conocer los aspectos generales del informe, así como los antecedentes en el cual se menciona una breve reseña de la empresa Interbank, la descripción de la organización en donde se indica el servicio brindado por la empresa, en la sección del contexto socioeconómico se narra el desarrollo de la empresa a nivel económico y social, las oportunidades, amenazas, una descripción general de la experiencia, funciones, los propósitos del puesto que ejercí, producto o proceso y los resultados obtenidos.

El segundo capítulo fundamenta las aplicaciones teóricas, los tipos de pruebas de software, así como no funcionales y funcionales, basadas en la experiencia, etc. También se desarrollan los principios generales de las pruebas, técnicas de pruebas, los métodos y procedimientos.

En el tercer capítulo, se desarrolló todo lo relacionado a la experiencia obtenida, se da a conocer aquellos aportes que se obtuvieron a raíz del conocimiento teórico adquirido en la etapa de estudiante en las aulas de Universidad Autónoma del Perú para finalizar con el informe se listan las conclusiones y recomendaciones brindadas, obtenidas a raíz del informe donde se evidencia las labores desempeñadas y experiencias obtenidas.

CAPÍTULO I

ASPECTOS GENERALES DEL TEMA

1.1. Antecedentes

Interbank (2019) comenta:

El Banco Internacional del Perú se fundó el 1 de mayo de 1897, e inició sus operaciones el 17 del mismo mes con un Directorio presidido por el Sr. Elías Mujica. Su primer local estuvo ubicado en la calle Espaderos, hoy Jirón de la Unión. En 1934 comenzó el proceso de descentralización administrativa, siendo Chiclayo y Arequipa las primeras agencias en abrirse, seguidas un año después por las de Piura y Sullana. La política de expansión se intensificó de manera notable en 1942, adquiriendo una propiedad en la Plazuela de la Merced y otra en la calle Lescano, donde se construyó el edificio "Sede La Merced", cuyos bellos acabados han llevado al Instituto Nacional de Cultura a catalogarlo como Monumento Histórico. En el pasado, el accionariado del banco incluyó capitales vinculados a la agroindustria, como La Fabril S.A. y W.R. Grace Co. En la década de los setenta, el Chemical Bank de Nueva York participó en el accionariado y fue responsable de la gerencia del banco. En 1970, el Banco de la Nación adquiere el mayor porcentaje de las acciones del banco, convirtiéndose de esta forma en miembro de la Banca Asociada del país. En 1980 el banco pasó a llamarse Interbank. El 20 de julio de 1994 un grupo financiero liderado por el Dr. Carlos Rodríguez-Pastor Mendoza, e integrado por grandes inversionistas como Nicholas Brady (ex secretario del Tesoro de los EEUU), entre otros, se convirtió en el principal accionista del banco al adquirir el 91% de las acciones disponibles. (párr. 1)

1.2. Descripción organizacional

Interbank forma parte del sector Financiero y Seguros; otorga servicios de préstamos, créditos, seguros, billetera digital. Con la finalidad de captar clientes no bancarizados, Interbank tiene en su respaldo a Tunki como una aplicación gratuita, Tunki es una billetera digital en el cual también permite vincular la tarjeta Interbank y así realizar operaciones como plin, recargas de celular, pago de servicios.

Visión

La visión de la organización manifiesta lo siguiente:

“Ser el mejor banco a partir de las mejores personas” (Interbank, 2019, párr. 2).

Propósito

El propósito de la organización es el siguiente:

“Acompañamos a los peruanos a alcanzar sus sueños, hoy” (Interbank, 2019, párr. 3).

Valores

Dentro de la organización Interbank se vive al máximo los valores:

“Integridad, Colaboración, Coraje, Innovación, Pasión por el servicio y sentido del humor.” (Interbank, 2019 , párr 4)

1.3. Contexto socioeconómico

Se realizó el análisis FODA en base a lo social y económico de la organización.

1.3.1. Oportunidades

- a.) Social: Crecimiento de pequeñas y medianas empresas
- b.) Económico: incrementar la cartera de clientes bancarizados y no bancarizados (Billetera digital Tunki)

1.3.2. Amenazas

- a.) Social: Pandemia, estado de emergencia, nueva ola del Covid.
- b.) Económico: Competencia entre otras entidades financieras o cajas.

1.4. Descripción general de experiencia

Software Test Engineer que Lidera y forma parte del equipo de certificación de la crew Transformación Tunki en uno de sus squads conformado por desarrolladores backend, mobile, Qa, Product owner y scrum master bajo la metodología Scrum.

1.5. Explicación del cargo, funciones ejecutadas

El cargo ocupado en la empresa Interbank es de Software Test Engineer y las funciones se mencionan a continuación:

- Definir estrategia de pruebas
- Seguimiento y ejecución pruebas funcionales/automatizadas/integración/regresión
- Elaboración de análisis y diseño de casos de prueba
- Elaboración de plan de Pruebas
- Elaboración los informes de los resultados de prueba con las recomendaciones correspondientes.
- Seguimiento y control al proceso de certificación
- Participación en las estimaciones de las historias de usuario
- Seguimiento y creación de defectos en ambiente pre productivo
- Participar en la estrategia de implantación de un release en producción
- Participación en las ceremonias ágiles como planning, review, daily y retrospectiva
- Coordinación con otras áreas
- Gestión con el equipo de calidad y metodología.

- Implementación de pruebas automatizadas, creación de scripts para el módulo de la OIM.
- Brindar soporte y capacitar sobre las metodologías de gestión de proyectos y peticiones.
- Gestión en la implementación de revisión de código (SonarQube) e integración con Jenkins.

1.6. Propósito del puesto

Un Software Test Engineer en la organización tiene como principal propósito asegurar la calidad del producto, estableciendo las estrategias de pruebas respectivas desde ingresa como un requerimiento hasta la implantación del reléase en producción.

Los que desempeñan el puesto tienen una experiencia más de 3 años en la labor de Testing, considerando 2 años de experiencia laboral en pruebas para proyectos ágiles y 2 más en pruebas en Entidades Financieras, experiencia en mantenimiento y ejecución de scripts para pruebas automatizadas y funcional, experiencia en herramientas como BitBucket, GitHub, Jira, y de gestión de pruebas.

Habilidades de comunicación efectiva, capacidad de organización y afán por la investigación e innovación.

1.7. Producto o proceso

Tunki es un aplicativo Mobile del banco Interbank, disponible en los sistemas operativos iOS y Android. Tunki está orientado a las personas del sector no bancarizado, es por ello, con Tunki puedes abrir una billetera 100% digital gratis, es una excelente opción que permite acceder a los beneficios que ofrece Tunki sin necesidad de acercarte presencialmente a una agencia de Interbank para aperturar una cuenta simple sin ningún costo de mantenimiento.

Tunki cuenta con un sistema biométrico rápido, con una mejor experiencia hacia el usuario y sobre todo seguro que permitirá realizar operaciones como transferencias PLIN, pago de servicios, recarga de celular, pagos QR Izipay/Niubiz y retiro sin tarjeta.

1.8. Resultados

Los resultados se han visto reflejados en:

- a.) Restablecer contraseña:** Con la implementación de la biometría facial en el flujo de restablecer contraseña se garantiza la seguridad de la cuenta de todos los usuarios ya que este selfie que se le solicita al usuario internamente realiza una validación de la foto selfie tomada desde el aplicativo con la foto almacenada en RENIEC, como resultado se obtiene un porcentaje de similitud entre las dos fotos y si este porcentaje de similitud se encuentra en el rango esperado quiere decir que se trata del mismo usuario que intenta restablecer una nueva contraseña y por lo tanto culmina con la creación de la nueva contraseña.
- b.) Tiempo de respuesta:** Con esta implementación de biometría facial disminuyó a 30 segundos en realizar el cambio de contraseña. Anteriormente el tiempo promedio que el usuario empleaba al restablecer contraseña era de 1 minuto como mínimo y hasta 5 minutos, esto se debía a que en el anterior flujo se le enviaba una clave al correo electrónico del usuario y muchas veces tardaba en llegar la clave o el usuario no podía acceder a su correo electrónico, esta clave tenía un tiempo de vigencia y si vencía el usuario tenía que volver a iniciar todo el flujo.
- c.) Uso de la aplicación:** En el flujo anterior era una limitante para el usuario no tener acceso a su correo electrónico ya que ahí se le enviaba una clave para

que logre cambiar su contraseña, esto ocasionaba que el usuario no pueda acceder a la aplicación para realizar otras operaciones, con esta implementación de biometría facial el usuario podrá restablecer su contraseña la veces que lo requiera y como resultado trajo que se realicen más transacciones como plin, pago de servicios, recargas.

CAPÍTULO II

FUNDAMENTACIÓN

2.1. Aplicaciones teóricas

Para el desarrollo de este trabajo se utilizó la norma ISO 9126 y Junta Internacional de Calificaciones de Pruebas de Software conocida como ISTQB.

2.1.1. *Pruebas del software*

Las pruebas del software tienen como finalidad la verificación del adecuado comportamiento de un programa a través de la selección de determinados casos de prueba orientados al resultado esperado del requerimiento.

La manera de ver a las pruebas de software ha cambiado de una manera más constructiva. No se cree que las pruebas de software recién inician cuando la fase de desarrollo ha culminado y el único fin es detectar defectos. Hoy en día el inicio de las pruebas de software puede estar presente en todo el desarrollo del software (Swebok, 2004).

Los defectos encontrados en las pruebas se le brinda la información necesaria al desarrollador, usualmente no llegan a corregir todo lo reportado pero la información que le brindamos debería ser de mucha ayuda para corregir como mínimo los defectos más críticos.

La realidad es que las pruebas deben ser ejecutadas para prevenir defectos, para esto todo el equipo debe estar involucrado en la implementación del programa. Las pruebas tempranas traen mucho beneficio entre actores que realizan las pruebas y los que desarrollan (Black & Rueda, 2011).

2.1.2. Principios generales de las pruebas

Existe siete principios:

a). Las pruebas delatan la existencia de defectos.

Las pruebas demuestran la existencia de incidencias en el sistema y no la ausencia, cuando se reporta un defecto, el desarrollador procede a corregirlo, con esta actividad es posible reducir la cantidad de incidencias, pero no la inexistencia de esta (Black & Rueda, 2011).

b). No es posible realizar pruebas exhaustivas.

Se cree que el objetivo del equipo de pruebas es solo asegurarse de que el software funcione correctamente antes de desplegarlo a producción, esto es imposible por diversas razones.

Por un lado, si consideramos los distintos caminos que existen para ejecutar un software de gran funcionalidad, esto puede resultar infinito ya que dentro de la lógica del software puede existir miles de sentencias y esto causa diversas combinaciones.

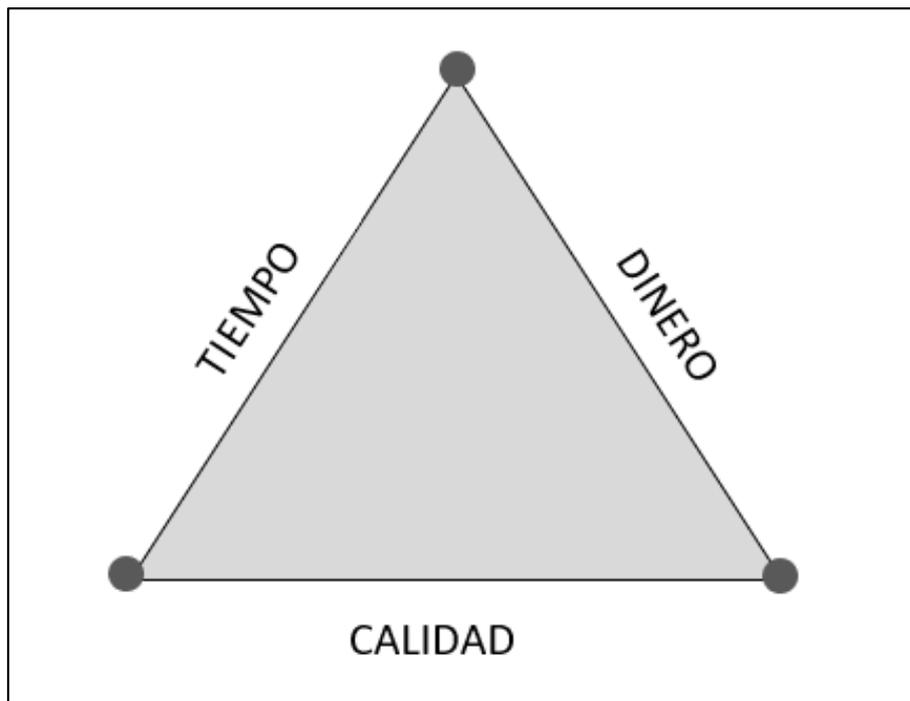
Por otro lado, si se corrige un defecto, por más que este sea puntual o de prioridad trivial, esto puede ocasionar que sea necesario realizar pruebas de regresión significativas e incluso puede estar aislado del cambio (Black & Rueda, 2011).

c). El beneficio de realizar pruebas tempranas.

Según la Figura 1, el costo de corregir un bug en el ambiente de producción es mucho mayor tanto monetario como el esfuerzo de la persona que realizará la corrección. Las pruebas tempranas nos ayudan a prevenir los defectos antes de la finalización del desarrollo del software (Black & Rueda, 2011).

Figura 1

Representación del costo de corregir un bug en el ambiente de producción



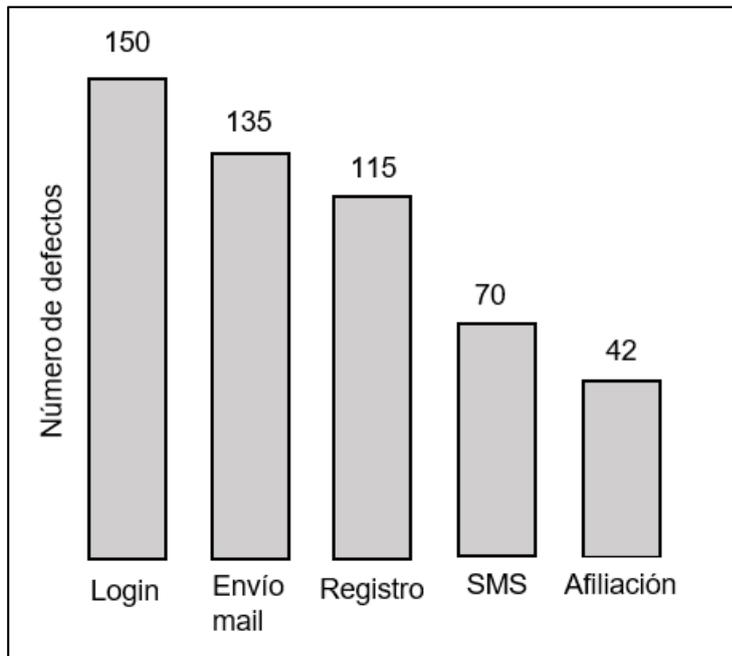
Nota: Ever Curiel (2021) Los beneficios más importantes de las Pruebas tempranas

d). Agrupar defectos.

Según la Figura 2, para agrupar los defectos podemos realizar un diagrama de Pareto, el análisis es sencillo, la información obtenida será valiosa ya que nos permite guiar las pruebas. La agrupación puede ser por comportamiento, componente, funcionalidad, etc. Pocas organizaciones realizan este tipo de análisis a pesar de que es sencillo de obtener (Black & Rueda, 2011).

Figura 2

Representación gráfica de la agrupación de defectos por cantidad y por funcionalidad



Nota: Black & Rueda., 2011, Agrupar defecto.

e). La paradoja del pesticida.

Ejecutar de los mismos casos varias veces dará como consecuencia encontrar menos defectos, no se encontrarán nuevos defectos. Por ejemplo, si siempre hacemos pruebas funcionales no es posible encontrar defectos de otros tipos de prueba como la de rendimiento. Así que se debe apostar por otras estrategias de pruebas (Black & Rueda, 2011).

f). Las pruebas deberían adaptarse a necesidades específicas.

Este es el último principio, las pruebas deben adaptarse a la necesidad y situación de la organización, proyecto y producto. Muchos ejecutan la prueba solo enfocándose en el cambio sin ver el panorama completo y aplicando buenas prácticas de otros proyectos (Black&Rueda, 2011).

g). La falacia de la ausencia de errores.

Encontrar muchos defectos y que estos defectos sean resueltos o encontrar pocos defectos no garantiza un resultado satisfactorio. Existen muchos productos en el mercado con pocos defectos que no han obtenido los resultados esperados en comparación de otros productos con mayor cantidad de defectos que, sin embargo, tienen otras bondades significativas para el equipo de negocio (Black & Rueda, 2011).

2.1.3. Técnicas de pruebas

Se tiene 2 tipos de pruebas, estáticas y dinámicas.

2.1.4. Pruebas dinámicas

Uno de los objetivos de esta prueba es la detección del defecto:

Las pruebas dinámicas son aquellas que se realizan mientras el código está en ejecución. Tienen como objetivo asegurar que el software se comporte de acuerdo con los requerimientos del negocio mediante la realización de pruebas funcionales y no funcionales. Estas pruebas se enfocan en la detección y confirmación de la corrección de defectos en el software. Por lo general se realizan en una etapa más tardía que las pruebas estáticas, por lo cual, los defectos encontrados en estas son más costosos. (Gomez, 2021, párr. 1)

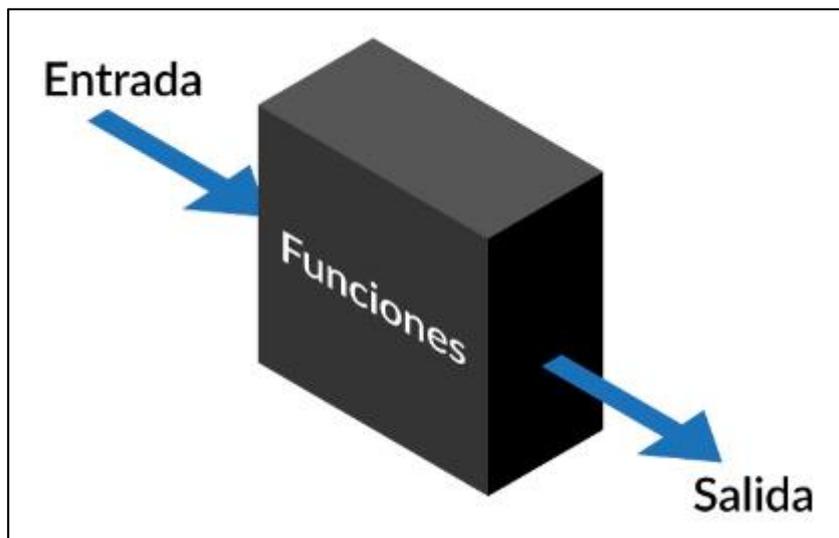
Dentro de las pruebas dinámicas tenemos los siguientes tipos:

a). Pruebas de caja negra.

Según la Figura 3, también son llamadas pruebas de comportamiento, son basadas en requerimientos funcionales del software, tienen como uno de sus objetivos encontrar las siguientes categorías de errores: 1) errores en el front, 2) errores funcionales, 3) errores en la base de datos (Pressman, 2015).

Figura 3

Interpretación gráfica de las pruebas de caja negra



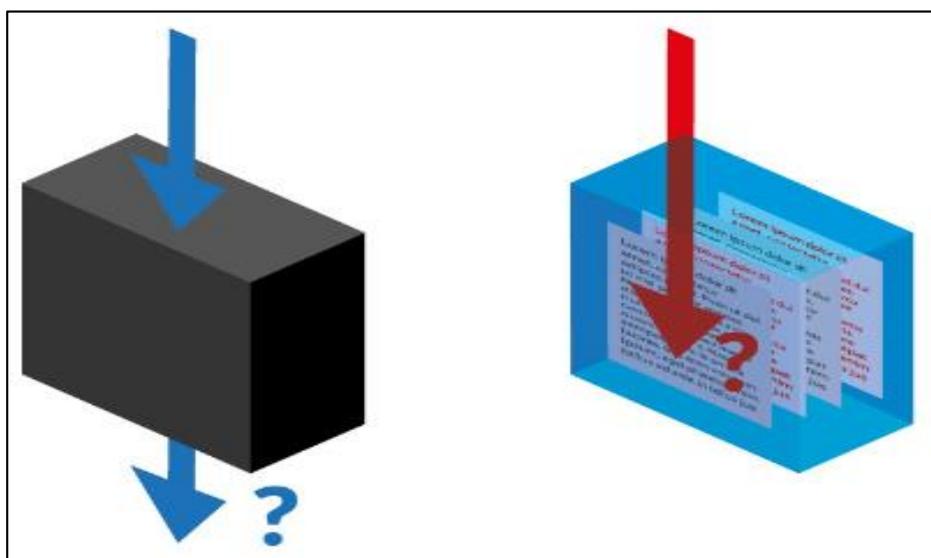
Nota: Desarrollo e implementación de aplicaciones web pruebas de caja negra

b). Pruebas de caja blanca.

Según la Figura 4, las pruebas de caja blanca se definen como la técnica de analizar el código, el diseño, estructura con la finalidad de garantizar la seguridad y eficiencia del sistema (Keedcoding, 2022, párr 2).

Figura 4

Interpretación gráfica de las pruebas de caja blanca



Nota: Desarrollo e implementación de aplicaciones web pruebas de caja negra.

2.1.5. Pruebas funcionales

Estas pruebas nos garantizan el correcto funcionamiento del software según los requerimientos indicados por el equipo de negocio. Las pruebas funcionales incluyen pruebas de regresión, pruebas unitarias, pruebas de integración, prueba de humo, prueba de aceptación, etc. (Consejos técnicos, 2020, párr. 2).

a). Pruebas de regresión.

Cuando el equipo de testing encuentra defectos en el software, el desarrollador corrige el defecto reportado, a raíz de esta actividad puede haber la posibilidad de que la solución dada haya causado otros defectos en el software, las pruebas de regresión son realizadas para asegurar que las correcciones realizadas no tengan impacto en negativo en la funcionalidad ya existente. El objetivo es encontrar defectos que fueron consecuencia de una solución (Vargas, 2022, párr. 8).

b). Pruebas unitarias.

Vargas (2022) menciona que:

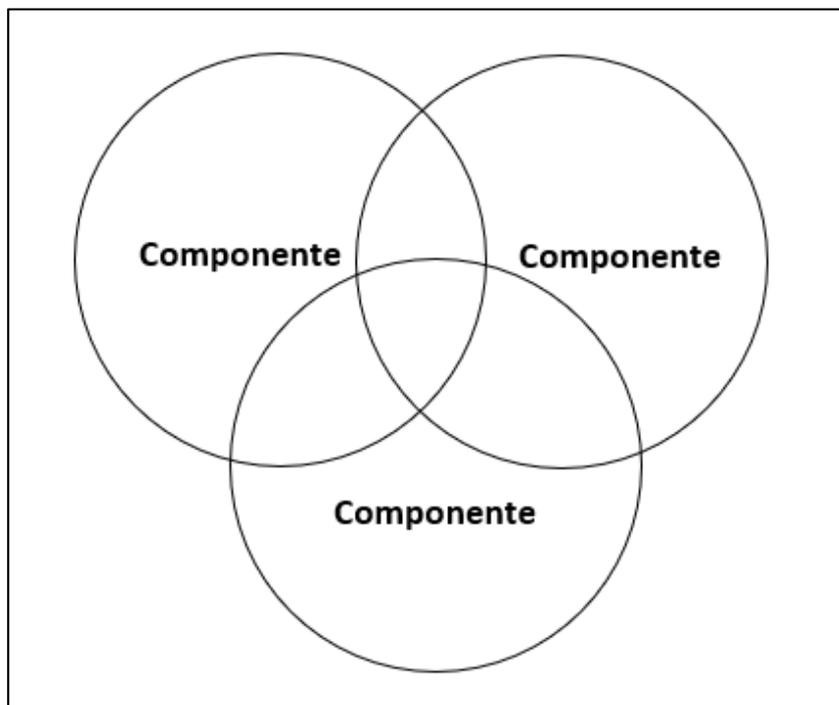
Las pruebas unitarias son las que aseguran que cada célula del código desarrollado en un componente brinde los resultados adecuados. En estas pruebas los desarrolladores observan la interfaz y la especificación de un componente, proporcionando la documentación del desarrollo del código se prueba exhaustivamente, claro que de forma independiente antes de pasar a otra unidad. Las pruebas unitarias son las que aseguran que cada célula del código desarrollado en un componente brinde los resultados adecuados. En estas pruebas los desarrolladores observan la interfaz y la especificación de un componente, proporcionando la documentación del desarrollo del código se prueba exhaustivamente, claro que de forma independiente antes de pasar a otra unidad. (párr. 3)

c). Pruebas de integración.

Según la Figura 5, esta prueba es una de las más comunes, es realizada para probar los componentes individuales los cuales luego de integrarse conformarán un módulo. “Generalmente se identifican errores en interfaz, al hacer una llamada a las APIS, base de datos, etc.” (Vargas, 2022, párr. 6).

Figura 5

Interpretación las pruebas de integración



Nota: Pruebas de integración.

d). Pruebas humo.

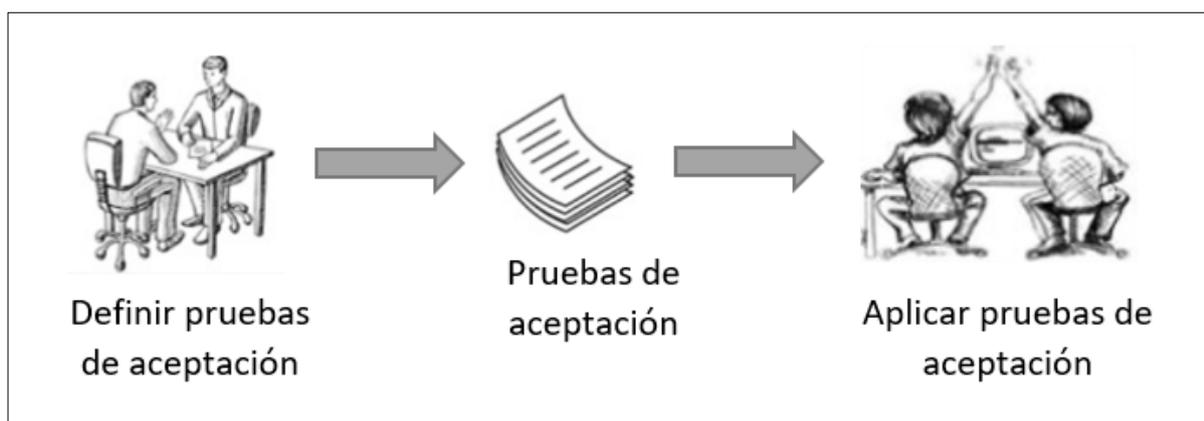
Es una de las pruebas más importantes, se verifican para asegurar que las funcionalidades de mayor impacto sigan funcionando correctamente, estas pruebas deben ser rápidas y sencillas, luego de una compilación esta prueba debería ser la primera que se debe realizar para saber que la compilación fue exitosa. (Vargas, 2022, párr. 5)

e). Pruebas de aceptación.

Estas pruebas son muy importantes ya que se define junto con el usuario cuáles serán las pruebas que él considera para verificar que cumplen con los requerimientos que establecieron inicialmente y brindar su conformidad tal y como se muestra en la Figura 6.

Figura 6

Definición de las pruebas de aceptación junto con el usuario que realizará las pruebas a la aplicación



Nota: Pruebas de aceptación

2.1.6. Pruebas no funcionales

Una perspectiva distinta a las pruebas automatizadas:

Las pruebas de software no funcionales son las que se hacen desde una perspectiva totalmente diferente a las pruebas automatizadas. Este tipo de plan de pruebas son un medio de control de calidad, que se realiza en aplicaciones de software para asegurarse de que todo funciona bien y poder saber en qué circunstancias podrían fallar. Las pruebas de software no funcionales son las que se hacen desde una perspectiva totalmente diferente a las pruebas automatizadas. Este tipo de plan de pruebas son un medio de control de calidad, que se realiza en aplicaciones de software para asegurarse

de que todo funciona bien y poder saber en qué circunstancias podrían fallar. (Soto, 2021, párr. 1)

Según la Tabla 1, dentro de las pruebas no funcionales tenemos:

Tabla 1

Tipo de pruebas no funcionales

Pruebas no funcionales
Pruebas de carga
Pruebas de rendimiento
Pruebas de estrés

2.1.7. Pruebas basadas en las experiencias

Depende mucho de las habilidades de la persona que realizará las pruebas, como su nombre lo dice, está basada en la experiencia con la tecnología, negocio, y aplicaciones.

2.1.8. Pruebas estáticas

Las pruebas estáticas previenen defectos:

A diferencia de las pruebas dinámicas, estas no requieren de la ejecución de software para ser realizadas. Parte del objetivo de las pruebas estáticas es la revisión de productos de trabajo como documentos de requerimientos, casos de prueba, planes de prueba, código, guías de usuario.

Estas pruebas se enfocan en la prevención de defectos y en la detección temprana de los mismos, ya que se pueden realizar en cualquier etapa del ciclo de vida de software según la información que se tenga disponible. (Gomez, 2021, párr. 2)

2.2. Métodos y procedimientos

Existe el siguiente estándar de la norma ISO/IEC 9126:2001:

2.2.1. Norma ISO/IEC 9126:2001

Verity (2022) manifiesta lo siguiente:

El estándar ISO/IEC 9126: 2001 presenta un marco conceptual para el modelo de calidad y define un conjunto de características y subcaracterísticas, las que debe cumplir todo producto software. El estándar ISO-9126 establece que cualquier componente de la calidad del software puede ser descrito en términos de una o más de las siete características básicas: funcionalidad, confiabilidad, usabilidad, eficiencia, mantenibilidad, portabilidad y satisfacción; cada una de ellas se detalla a través de un conjunto de subcaracterísticas que permiten profundizar en la evaluación de la calidad de productos de software.

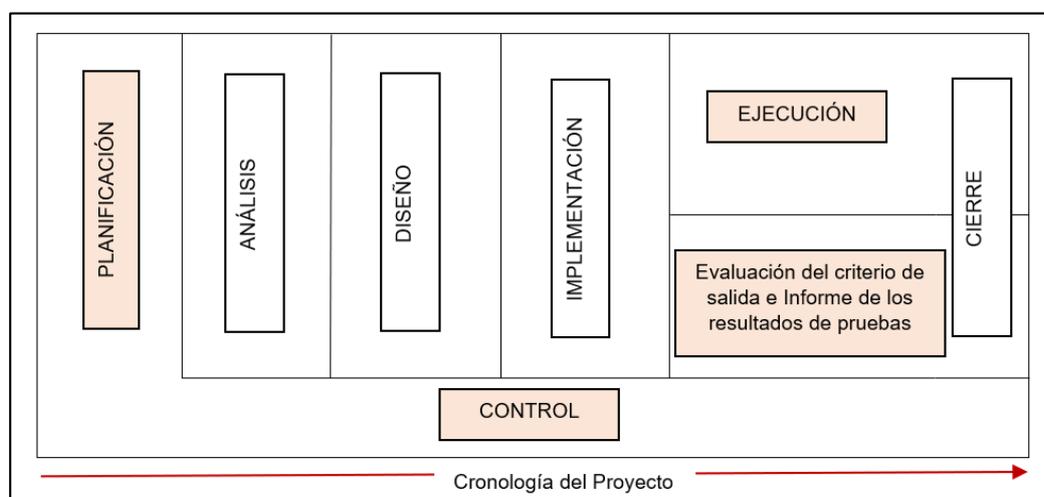
(párr. 2)

2.2.2. Proceso de pruebas del ISTQB

Según ISTQB el proceso de pruebas incluye los pasos siguientes según la Figura 7:

Figura 7

El proceso de pruebas de ISTQB y la cronología del proyecto



Nota: Black & Rueda. (2011) Proceso de software

a). Planificación y control.

La planificación y control consiste en actividades como determinar estrategias, alcance, riesgos y objetivos de las pruebas. Definir la cantidad de recursos necesarios, estrategia de pruebas, criterios de salida el cual conlleva actividades como el análisis de los resultados, monitoreo y toma de decisiones (Black & Rueda, 2011).

b). Análisis y diseño.

El análisis consiste en revisar el requerimiento, en base a ello identificar los casos de prueba de acuerdo con el alcance planificado incluyendo las especificaciones de diseño o requisitos, riesgos o arquitectura. En cuanto al diseño incluye actividades como identificar la data requerida para las pruebas, identificar las precondiciones de la prueba, así como los pasos que se debe seguir para ejecutar el caso de prueba y el resultado esperado (Black & Rueda, 2011).

c). Implementación y ejecución.

La implementación implica actividades principales como la priorización de las pruebas identificadas, creación de data de prueba y la redacción del procedimiento, en el caso de ser pruebas automatizadas esto implica la creación del script, orden de las pruebas y verificar que los cambios hayan sido desplegados en el ambiente donde se ejecutará las pruebas. Con respecto a la ejecución conlleva actividades como la ejecución de los casos de prueba ya sean automatizados o manuales, evidenciar los resultados de las pruebas indicando la versión donde se ha realizado la prueba, comparar el resultado real con lo esperado, reportar los defectos y el retest respectivo, ejecución de pruebas de regresión cuando se requiera (Black & Rueda, 2011).

d). Evaluar los criterios de salida, creación de informes y Cierre.

Incluye actividades de gestión como comprobar evaluar si fueron suficientes las pruebas realizadas, redactar el informe con el respectivo resumen de las pruebas hacia el equipo de negocio (Black & Rueda, 2011).

CAPÍTULO III

APORTES Y DESARROLLO DE EXPERIENCIAS

3.1. Aportes teóricos

De acuerdo con los conocimientos adquiridos en el transcurso de mis estudios en las aulas de la Universidad Autónoma del Perú, y con base en las teorías ya mencionadas y detalladas en los capítulos anteriores de este informe, se procedió a plantear las soluciones desarrolladas en los aportes prácticos.

3.1.1. *Objetivo general*

El proceso de Cambio de contraseña desde la aplicación de Tunki se realice mediante la biometría facial para brindar mayor seguridad a la cuenta del usuario.

3.1.2. *Realidad problemática*

El proceso en el que un usuario cambiaba su contraseña en el aplicativo era mediante el ingreso de su documento nacional de identidad (DNI), carné de extranjería (CE) o pasaporte (PAS); luego de ello se le enviaba un código a su correo electrónico registrado, debía ingresar el código en el aplicativo, una vez validado que este código es correcto el usuario procedía a crear la nueva contraseña y culminaba el flujo ingresando a la aplicación con la nueva contraseña.

La persona que tuviera acceso al correo electrónico podría cambiar la contraseña de un usuario. Para brindar mucha más seguridad al cambio de contraseña se ha implementado el flujo de “Biometría facial” el cual validará que la persona que quiera cambiar la contraseña sea a quién le pertenezca la cuenta.

3.2. Aportes prácticos

3.2.1. Refinamiento

En esta ceremonia ágil participó el equipo de Back, Mobile, Qa y Product Owner, la finalidad fue presentar la historia de usuario al equipo, el cual contenía la información del requerimiento. En el refinamiento el equipo realizó todas las consultas respectivas y también mencionaron las distintas casuísticas a considerar para el desarrollo de la historia de usuario.

En esta reunión también se elaboró en conjunto el example mapping el cual contiene las reglas de negocios con sus respectivos ejemplos.

En la Tabla 2, se muestra la primera regla de negocio “La opción ¿olvidaste tu clave? Debe llevar a la pantalla ‘Cambia tu clave’” con sus respectivos 2 ejemplos.

Tabla 2

Elaboración del example mapping con respecto a la regla 1 y su respectivo ejemplo con respecto a la historia de usuario

Regla 1	Ejemplo
La opción ¿olvidaste tu clave? Debe llevar a la pantalla “Cambia tu clave”.	El usuario se encuentra en el login y selecciona ¿olvidaste tu clave? entonces se le dirigirá el flujo de biometría.
	El usuario puede cambiar las veces que desee su contraseña.

En la Tabla 3, se muestra la primera regla de negocio “Solo los que tienen el tipo de documento DNI les mostrará el flujo de biometría facial y podrán cambiar la contraseña” con sus respectivos 2 ejemplos.

Tabla 3

Elaboración del example mapping con respecto a la regla 2 y su respectivo ejemplo con respecto a la historia de usuario

Regla 2	Ejemplo
Solo los que tienen el tipo de documento DNI les mostrará el flujo de biometría facial y podrán cambiar la contraseña.	<p>El usuario con DNI se encuentra en la pantalla cambia tu clave, al seleccionar siguiente debe mostrar el mensaje “Campo obligatorio”.</p> <p>El usuario con DNI se encuentra en la pantalla cambia tu clave, ingresa el # de su DNI entonces selecciona siguiente debe mostrarle la pantalla donde ingrese la OTP que le llegó como SMS.</p>

En la Tabla 4, se muestra la primera regla de negocio “Los que tienen algún tipo de documento CE y PAS deberán comunicarse con el centro de ayuda” con sus respectivos 2 ejemplos.

Tabla 4

Elaboración del example mapping con respecto a la regla 3 y su respectivo ejemplo con respecto a la historia de usuario

Regla 3	Ejemplo
Los que tienen tipo de documento CE y PAS deberán comunicarse con el centro de ayuda.	El usuario con CE se encuentra en la pantalla cambia tu clave, ingresa su # documento y selecciona siguiente entonces mostrará un pop up “cambia tu clave”.
	El usuario con PAS se encuentra en la pantalla cambia tu clave, ingresa su # documento y selecciona siguiente entonces mostrará un pop up “cambia tu clave”.

3.2.2. Planificación y control

En este paso se realizó la ceremonia del planning en el cual el equipo estimó el esfuerzo que le tomará desarrollar la historia de usuario.

En la Tabla 5, se detalla los aspectos generales elaboración de un plan de pruebas que contiene información importante de las pruebas que se realizó.

Tabla 5

Aspectos generales del plan de pruebas de historia de usuario

Título	Descripción
Alcance	Implementar biometría facial en recuperar contraseña
Fuera del alcance	Otras funcionalidades.
Ambiente de pruebas	UAT.
Hardware, software y dispositivos externos	Dispositivos Android y iOS.
Accesos	Base de datos, TestLink.
Data de prueba	Usuario con DNI, Carné de extranjería y Pasaporte
Aplicaciones	Tunki.
Conformidad a las pruebas	Product Owner

En la Tabla 6, se describe la estrategia de pruebas a realizar en las pruebas de certificación correspondiente a la historia de usuario.

Tabla 6*Estrategia de pruebas del plan de pruebas*

Estrategia de pruebas	Descripción
1	Las pruebas se realizan en el ambiente de UAT. Las pruebas se ejecutarán de acuerdo con las prioridades definidas en la HU, los criterios de aceptación y reglas definidas en el example mapping.
2	Las pruebas de certificación se realizan en los dispositivos iOS y Android.
3	Las ejecuciones de los casos de prueba se realizarán en TestLink, de igual manera las evidencias serán cargadas en la herramienta.
4	Antes de implantar el reléase en producción se debe realizar pruebas de humo, para esto el equipo deberá hacer un merge en back/front y desplegarlo en el ambiente de UAT.
5	Antes de implantar el reléase en producción se debe realizar pruebas de humo, para esto el equipo deberá hacer un merge en back/front y desplegarlo en el ambiente de UAT.
6	Las pruebas se realizarán en el ambiente productivo, estas pruebas estarán disponible solo para una cantidad limitada de usuarios.

En la Tabla 7, se indica los tipos de pruebas a realizar en las pruebas de certificación correspondiente a la historia de usuario.

Tabla 7

Tipos de pruebas del plan de pruebas

Tipos de prueba	Descripción
Funcionales	Integración. Regresión. Humo.
No funcionales	Carga.

3.2.3. Análisis y diseño - sprint 1

En este paso el equipo de certificación realizó ciertas tareas para analizar y poder identificar los casos a probar, luego de ello se realiza la revisión por pares con los integrantes del equipo para asegurar la calidad de los casos.

Una vez que se identificó los casos a probar, se procedió a iniciar con el diseño de los casos de prueba, tal y como se muestra en la Figura 8. Este se realiza en un formato Excel macro el cual luego fue importado en la herramienta de gestión de pruebas “TestLink”.

Figura 8

Modelo del formato para diseñar los casos a ejecutar

INVENTARIO DE CASOS DE PRUEBA							
NUM casos de prueba		*Consulta las Instrucciones	Generar	*Consulta las Instrucciones	Mostrar / Ocultar Campos de Carga Masiva		
Identificación y clasificación				Especificación detallada			
Nivel de Prueba	<Nivel de Agrupación >	Nombre del caso de prueba	Criticidad	Descripción	Precondiciones	Pasos	Resultados Esperados
Pruebas de Sistema-Funcionales	[NF] Recuperar contraseña con biometría	CP001 Validar que al estar en la pantalla escribe el código SMS cuando presione en Continuar muestra pantalla Ahora validaremos tu identidad	Alta	Validar que al estar en la pantalla escribe el código SMS cuando presione en Continuar muestra pantalla Ahora validaremos tu identidad	Given que soy usuario Tunki And tener los permisos de camara desactivada	Abrir el app Tunki Presionar en ¿Olvídate clave? ingresar número de documento y presionar enviar Ingresar el código y presionar en continuar	Mostrara Login mostrar pantalla de Cambia tu clave mostrar pantalla de Escribe el código que te hemos enviado Mostrar pantalla de Ahora validaremos tu identidad con un sello Mostrara Login
Pruebas de Sistema-Funcionales	[NF] Recuperar contraseña con biometría	CP002 Validar que al estar en la pantalla Ahora validaremos tu identidad cuando se presione en el boton Empezar muestra pop up de permisos de camara	Alta	Validar que al estar en la pantalla "Ahora validaremos tu identidad" cuando se presione en el boton Empezar muestra pop up de permisos de camara	Given que soy usuario Tunki And tener los permisos de camara desactivada	Abrir el app Tunki Presionar en ¿Olvídate clave? ingresar número de documento y presionar enviar Ingresar el código y presionar en continuar Presionar en empezar	Mostrara Login mostrar pantalla de Cambia tu clave mostrar pantalla de Escribe el código que te hemos enviado Mostrar pantalla de Ahora validaremos tu identidad con un sello Mostrar popup solicitando permisos de camara Mostrara Login
Pruebas de Sistema-Funcionales	[NF] Recuperar contraseña con biometría	CP003 Validar que al estar en la pantalla Ahora validaremos tu identidad cuando se presione en el boton Empezar muestra pantalla para tomarse foto selfie	Alta	Validar happy path de creacion de usuario cuando acepta los permisos de autocompletado con PASS para android	Given que soy usuario Tunki And tener los permisos de camara activada	Abrir el app Tunki Presionar en ¿Olvídate clave? ingresar número de documento y presionar enviar Ingresar el código y presionar en continuar Presionar en empezar	Mostrara Login mostrar pantalla de Cambia tu clave mostrar pantalla de Escribe el código que te hemos enviado Mostrar pantalla de Ahora validaremos tu identidad con un sello Mostrar pantalla para tomarse la selfie

A continuación, en la Tabla 8, se muestran los detalles que contempló el diseño de casos de prueba.

Tabla 8*Información del diseño de caso de prueba*

Nombre del caso de prueba	Criticidad	Descripción	Precondiciones
CP002 Validar que al estar en la pantalla Ahora validamos tu identidad cuando se presione en el botón Empezar muestra pop up de permisos de cámara.	Alta.	Validar que al estar en la pantalla "Ahora validamos tu identidad" cuando se presione en el botón Empezar muestra pop up de permisos de cámara.	Tener los permisos de cámara desactivada.

En la Tabla 9, se muestran los detalles del paso a paso que se debe seguir para ejecutar un caso de prueba.

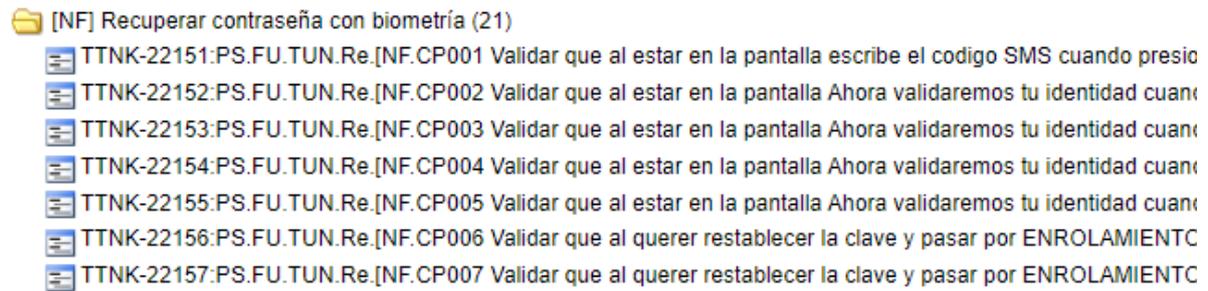
Tabla 9*Información de los pasos del caso de prueba 02*

Nivel	Paso	Resultado esperado
1	Abrir la aplicación de Tunki.	Mostrará Login.
2	Presionar en ¿Olvidaste clave?	mostrar pantalla de Cambia tu clave.
3	Ingresar el número de documento y presionar en continuar.	Mostrar pantalla "Escribe el código que te hemos enviado"
4	Ingresar el código y presionar en continuar.	Mostrar pantalla de ahora validaremos tu identidad con un selfie.
5	Presionar en empezar.	Mostrar pop up solicitando permisos de cámara.

Una vez finalizado el diseño de casos de prueba se procedió a crear plan de pruebas y el Build en TestLink para luego el importar los casos de prueba, así como se muestra en la Figura 9.

Figura 9

Casos de prueba importado en la herramienta Testlink

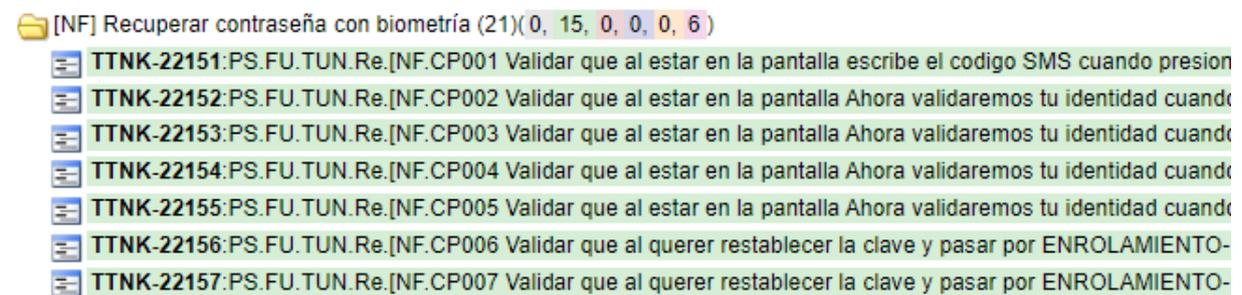


3.2.4. Implementación y ejecución

En este paso se realizó el despliegue de los cambios en UAT y la ejecución de casos de prueba, así como se muestra en la Figura 10.

Figura 10

Casos de prueba ejecutados en la herramienta Testlink



a). Despliegue de los cambios a UAT.

En este punto se gestionó el despliegue de los microservicios de back en UAT, para el caso de Android se generó un APK, este es un archivo que empaqueta la aplicación, en el caso de iOS se generó un IPA, IPA es un formato que Apple utiliza y sus siglas que significan iOS AppStore Package,

b). Creación de data.

Según el alcance de las pruebas de la historia de usuario, se creó la siguiente data para la ejecución de pruebas:

- Usuarios con registro en TUNKI con el tipo de documento DNI.
- Usuarios con registro en TUNKI con el tipo de documento CE.

- Usuarios con registro en TUNKI con el tipo de documento PAS.

c). Ejecución de casos de prueba.

Se procedió a realizar la asignación de los casos de prueba a las personas que lo ejecutaron en los sistemas operativos Android y iOS.

Existen estados en el que puede estar un caso de prueba, tal y como se muestra en la Figura 11, se tiene los estados Pasado, Desestimado y Fallido.

Figura 11

Estado de los casos de prueba

Fecha	Testeado por	Estado	Ejec (min)	Versión	Modo de ejecución
14/11/2022 23:47:49	asaravia	Pasado		1	
14/11/2022 23:47:45	asaravia	Desestimado		1	
14/11/2022 23:46:49	asaravia	Pasado		1	
15/09/2022 10:31:24	EricsonLN	Pasado	10.00	1	
- TUNKI_TUNKI-5104_CP001_Recuperar clave biometria.docx (462144 bytes, application/vnd.openxmlformats-officedocument.wordprocessingml.document) 15/09/2022					
15/09/2022 09:16:22	EricsonLN	Fallido		1	

En la Tabla 10 se detalla un abreve descripción de los estados que puede tener el caso de prueba.

Tabla 10

Estados de la ejecución de los casos a ejecutar

Estados	Descripción
Pasado	Cuando el CP ejecutado fue exitoso, cumplió con el resultado esperado.
Fallido	Cuando el caso de prueba ejecutado falló, el resultado esperado no fue satisfactorio.
Desestimado	Se realiza cuando el caso de prueba ya no es parte del alcance o no se puede replicar en el ambiente.

Existen estados en el que puede estar un caso de prueba, tal y como se muestra en la Figura 11, se tiene los estados Pasado, Desestimado y Fallido

d). Reportar los defectos.

Los defectos encontrados se registraron en la herramienta Jira y tuvo la siguiente estructura mencionada en la Tabla 11.

Tabla 11
Información que debe contemplar un defecto

Título	Descripción
Versión	Se indicó la versión del APK o IPA donde replicó el defecto. Se dio una breve descripción del defecto.
Descripción	
Resultado actual	Se explicó lo que actualmente está ocurriendo, se indicó el paso a paso hasta llegar a replicar el defecto y también se adjuntó videos o imágenes como evidencia.
Resultado esperado	Se redactó cual debe ser el correcto funcionamiento, lo que se espera de la prueba.
Asignación	Se asignó el defecto al desarrollador responsable de la corrección (Back, Mobile iOS o Android).

e). Retest.

Una vez que el defecto reportado fue solucionado, el desarrollador generó una nueva versión en mobile, cuando se corrige un defecto en back se realizan un despliegue en UAT. En ocasiones se realizó las pruebas de regresión para asegurar que la corrección del defecto no ocasionó otros defectos.

f). Evidencias de las pruebas.

Como se visualiza en la Figura 12, como parte del proceso de certificación, se evidenció cada caso de prueba, generando un documento que contiene los pantallazos como evidencia de la prueba. Este se adjuntó en la herramienta de gestión de pruebas.

Figura 12

Gráfica de la evidencia adjunta en Testlink



3.2.5. Evaluación de los criterios de salida, creación de informes y cierre

a). Sprint Review.

Luego de finalizar con las pruebas se procedió a agendar el sprint review, en esta ceremonia se invitó al equipo de TI y negocio para mostrarles una presentación del desarrollo finalizado en tiempo real.

b). Informe de pruebas.

Se procedió a elaborar el documento de “informe de pruebas de certificación”, el cual contiene información importante, así como información general tal como indica en la Tabla 11, estadísticas de ejecución, resultado de pruebas no funcionales si es que lo hubiese, hallazgos encontrados con su respectivo estado actual.

Tabla 11

Datos de la certificación en el informe de pruebas correspondiente a la historia de usuario

Prueba no funcional	Descripción
Id Jira	TUNKI-123
Aplicación	TUNKI
Historia de usuario	Restablecer clave con biometría facial
Plataforma de pruebas	Android / iOS
Responsable QA	Ana Saravia

Como se visualiza en la Figura 13, el resumen de la ejecución contiene información como la cantidad de casos ejecutados sin defectos, completados con defectos y también los casos desestimados.

Figura 13

Resumen de ejecución de casos de prueba

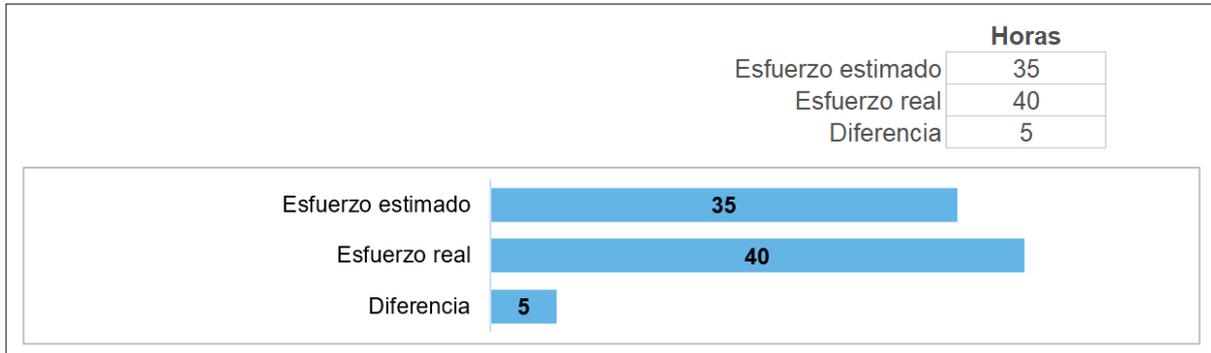


Como se visualiza en la Figura 14, se tiene una sección de estadística de esfuerzo de certificación, el cual contiene información como el esfuerzo estimado,

esfuerzo real y la diferencia que vendría a ser la resta del esfuerzo estimado menos el esfuerzo real.

Figura 14

Estadísticas del esfuerzo de certificación



En la Tabla 12 se indica si es que se contemplaron pruebas no funcionales en las pruebas realizadas

Tabla 12

Resultado de pruebas no funcionales

Prueba no funcional	Planificación
Stress	No requiere esta prueba
Carga	No requiere esta prueba

La Figura 15 representa la conformidad y no conformidad del resultado de las pruebas de certificación con respecto a la historia de usuario a desarrollar.

Figura 15

Valoración de las pruebas

5. Valoración final
El resultado final es

Conforme

Conforme

No conforme

En la Tabla 13 se detalla información importante de los defectos encontrados en las pruebas realizadas.

Tabla 13

Defectos encontrados en las pruebas

Nro de Issue	Descripción	Responsable	Prioridad	Estado
JIRA-001	No se visualiza el mensaje adecuado	Pedro Rios	Menor	Cerrado

Se solicitó la conformidad del Product Owner, con ello iniciamos la gestión del pase a producción.

c). Cierre de pruebas.

El equipo de desarrollo actualizó las ramas certificadas con la versión de producción (master), realizó la compilación y generó la nueva versión con el cual se realizó las respectivas pruebas de humo que fueron satisfactorias.

d). Despliegue del release en producción.

Se realizó el despliegue en una versión productiva para un grupo cerrado de pruebas, entre ellos participó el equipo de negocio quien debe realizar las validaciones correspondientes y luego el Product Owner brindar la conformidad del release desplegado en producción.

CONCLUSIONES

A través del este informe se llegó a las conclusiones siguientes: Se observó que el proceso de calidad de software debe estar inmerso en el proceso de desarrollo del producto, para esto fue importante definir una estrategia de pruebas y analizar el tipo de pruebas que se requiere para este tipo de implementación.

La calidad de software es responsabilidad de todo el equipo (producto owner, desarrolladores y Qa), desde el momento en el que el Product Owner da a conocer la iniciativa al equipo en los refinamientos. Las consultas o preguntas que surgen enriquecen la definición del comportamiento de la nueva funcionalidad con el fin de que no quede dudas al respecto y poder asegurar el correcto funcionamiento del cambio de contraseña con biometría facial en el ambiente de producción.

Los usuarios que por diferentes motivos no han logrado restablecer su contraseña y esto les ha impedido acceder al aplicativo para realizar transacciones, con la biometría facial podrán realizarlo sin problemas.

Se observó que la seguridad en un aplicativo genera mucha confianza entre sus usuarios y trae como consecuencia que más personas lo usen.

RECOMENDACIONES

Se recomienda realizar pruebas de calidad de software en todo el proceso de desarrollo del producto.

Según corresponda, se recomienda que el usuario realice sus pruebas en el ambiente de producción cerrado, esto significa que se le habilitará las pruebas solo a una cantidad limitada de usuarios con el fin que puedan realizar las pruebas requeridas según los criterios de aceptación definidas en la historia de usuario y así brinde su conformidad para el despliegue a producción.

Se recomienda utilizar un factor de seguridad como la biometría facial, una tecnología capaz de identificar a una persona a través de una imagen.

REFERENCIAS

- Black, R. & Rueda, G. (2011a). Pruebas del software.
<https://es.scribd.com/document/362234434/Fundamentos-de-pruebas-de-software-pdf>
- Black, R. & Rueda, G. (2011b). *Las pruebas delatan la existencia de defectos.*
<https://es.scribd.com/document/362234434/Fundamentos-de-pruebas-de-software-pdf>
- Black, R. & Rueda, G. (2011c). *No es posible realizar pruebas exhaustivas.*
<https://es.scribd.com/document/362234434/Fundamentos-de-pruebas-de-software-pdf>
- Black, R. & Rueda, G. (2011d). *El beneficio de realizar pruebas tempranas.*
<https://es.scribd.com/document/362234434/Fundamentos-de-pruebas-de-software-pdf>
- Black, R. & Rueda, G. (2011e). *Agrupar defectos.*
<https://es.scribd.com/document/362234434/Fundamentos-de-pruebas-de-software-pdf>
- Black, R. & Rueda, G. (2011f). *La paradoja del pesticida.*
<https://es.scribd.com/document/362234434/Fundamentos-de-pruebas-de-software-pdf>
- Black, R. & Rueda, G. (2011g). *Las pruebas deberían adaptarse a necesidades específicas.* <https://es.scribd.com/document/362234434/Fundamentos-de-pruebas-de-software-pdf>
- Black, R. & Rueda, G. (2011h). *La falacia de la ausencia de errores.*
<https://es.scribd.com/document/362234434/Fundamentos-de-pruebas-de-software-pdf>

Black, R. & Rueda, G. (2011i). *Planificación y control.*

<https://es.scribd.com/document/362234434/Fundamentos-de-pruebas-de-software-pdf>

Black, R. & Rueda, G. (2011j). *Análisis y diseño.*

<https://es.scribd.com/document/362234434/Fundamentos-de-pruebas-de-software-pdf>

Black, R. & Rueda, G. (2011k). *Implementación y ejecución.*

<https://es.scribd.com/document/362234434/Fundamentos-de-pruebas-de-software-pdf>

Black, R. & Rueda, G. (2011l). *Evaluación de los criterios de salida, creación de informes y Cierre.* <https://es.scribd.com/document/362234434/Fundamentos-de-pruebas-de-software-pdf>

Consejos técnicos, Pruebas de rendimiento. (2020). *Pruebas funcionales.*

<https://www.loadview-testing.com/es/blog/tipos-de-pruebas-de-software-diferencias-y-ejemplos/>

Gomez, C. (2021). *Pruebas Dinámicas.* <https://www.diariodeqa.com/post/pruebas-din%C3%A1micas-vs-pruebas-est%C3%A1ticas>

Interbank. (2019a). *Nuestra Historia.* <https://interbank.pe/nosotros>

Interbank. (2019b). *Vision.* <https://interbank.pe/nosotros1>

Interbank. (2019c). *Propósito.* <https://interbank.pe/nosotros1>

Interbank. (2019d). *Nuestros Valores.* <https://interbank.pe/nosotros1>

Keedcoding. (2022). *¿Qué son las pruebas de caja blanca?*

<https://keepcoding.io/blog/que-son-pruebas-de-caja-blanca/>

Pressman. (2015). *Pruebas de caja negra*.

<https://whyphi.staff.telkomuniversity.ac.id/files/2016/01/ebook-pressman-sw-engineering.pdf>

Swebok. (2004). *Guide to the Software Engineering Body of Knowledge*.

<http://www.cc.uah.es/drg/b/HispaSWEBOOK.Borrador.pdf>

Soto, V. (2021). *¿Qué son pruebas no funcionales (performance)?*

<https://www.pragma.com.co/blog/conoce-que-son-las-pruebas-no-funcionales-de-software>

Vargas, C (2022a). *Pruebas unitarias*. [https://trycore.co/transformacion-digital/tipos-](https://trycore.co/transformacion-digital/tipos-de-pruebas-funcionales/)

[de-pruebas-funcionales/](https://trycore.co/transformacion-digital/tipos-de-pruebas-funcionales/)

Vargas, C (2022b). *Pruebas de integración*. [https://trycore.co/transformacion-](https://trycore.co/transformacion-digital/tipos-de-pruebas-funcionales/)

[digital/tipos-de-pruebas-funcionales/](https://trycore.co/transformacion-digital/tipos-de-pruebas-funcionales/)

Vargas, C. (2022c). *Pruebas de humo*. [https://trycore.co/transformacion-digital/tipos-](https://trycore.co/transformacion-digital/tipos-de-pruebas-funcionales/)

[de-pruebas-funcionales/](https://trycore.co/transformacion-digital/tipos-de-pruebas-funcionales/)

Verity. (2022). *¿Qué es la norma ISO/IEC 9126: 2001?* [https://www.verity.cl/que-es-](https://www.verity.cl/que-es-norma-iso-iec-9126)

[norma-iso-iec-9126](https://www.verity.cl/que-es-norma-iso-iec-9126)

[2001/#:~:text=El%20est%C3%A1ndar%20ISO%2D9126%20establece,tra%20](https://www.verity.cl/que-es-norma-iso-iec-9126)

[C3%A9s%20de%20un%20conjunto%20de](https://www.verity.cl/que-es-norma-iso-iec-9126)

ANEXOS

Anexo 1. Constancia de Trabajo



Lima, 27 de Octubre del 2022

*Señores
interbank
Presente.-*

De mi consideración:

Por medio de la presente le informamos que el/la señor(a) SARAVIA HUAMANI, ANA MARIBEL es colaborador de nuestra empresa INTERBANK, desde el 11 de Enero del 2021, desempeñándose a la fecha en el cargo de SOFTWARE TEST ENGINEER 2 en la división de GCIA.CENTRAL ESTRATEGIA Y TRANSFORMACIÓN TECNOLÓGICA

Se extiende la presente a solicitud del Colaborador interesado, para los fines que estime conveniente.

Atentamente

División de Gestión y Desarrollo Humano

Handwritten signature of Julio del Valle Montero in black ink.

Julio del Valle Montero
Gerente División Gestión y Desarrollo Humano

Handwritten signature of Nancy Zambrano Norabuena in black ink.

Nancy Zambrano Norabuena
SubGerente Asesoría Experta GDH