



Universidad Nacional  
**SAN LUIS GONZAGA**



### **Atribución-NoComercial-SinDerivadas 4.0 Internacional**

Esta licencia es la más restrictiva de las seis licencias principales Creative Commons, permitiendo a otras solo descargar sus obras y compartirlas con otras siempre y cuando den crédito, pero no pueden cambiarlas de forma alguna ni usarlas de forma comercial.

<http://creativecommons.org/licenses/by-nc-nd/4.0>



UNIVERSIDAD NACIONAL "SAN LUIS GONZAGA"  
FACULTAD DE INGENIERÍA DE SISTEMAS  
DIRECCIÓN DE INVESTIGACIÓN  
EVALUACIÓN DE ORIGINALIDAD



## CONSTANCIA

El que suscribe, deja constancia que se ha realizado el análisis con el software de verificación de similitud al documento cuyo título es:

### INTERVENCION DEL QUALITY ASSURANCE EN LA MEJORA DE LA CALIDAD DEL APLICATIVO TUNKI DEL BANCO INTERBANK

Presentado por:

- **QUISPE GARAYAR LEONELA ALEJANDRA**

**BACHILLER** en **PREGRADO** de la facultad de Ingeniería de Sistemas. El resultado obtenido es (**porcentaje de similitud 0%**) por el cual se otorga el calificativo de:

**APROBADO**, según el Reglamento de Evaluación de la Originalidad.

Se adjunta al presente el reporte de evaluación con el software de verificación de originalidad.

Ica, 05 de abril de 2024

  
**Dr. JAVIER ORLANDO GUTIÉRREZ FERREYRA**  
Director de la Unidad de Investigación  
Facultad de Ingeniería de Sistemas

UNIVERSIDAD NACIONAL "SAN LUIS GONZAGA"

VICERRECTORADO DE INVESTIGACIÓN

FACULTAD Ingeniería de Sistemas



**INTERVENCION DEL QUALITY ASSURANCE EN LA MEJORA  
DE LA CALIDAD DEL APLICATIVO TUNKI DEL BANCO  
INTERBANK**

Línea de investigación: Ciencias naturales, ingeniería y tecnologías sostenibles

Autor: Bachiller Leonela Alejandra Quispe Garayar

Asesor: Erwin Peña Casas

**INFORME FINAL DE TRABAJO DE SUFICIENCIA PROFESIONAL  
PARA OPTAR EL TITULO DE INGENIERO DE SISTEMAS**

**Ica, Perú**

**2024**

## **DEDICATORIA**

El presente trabajo va dedicado especialmente a mis padres, que han sido mi guía y mi ejemplo a seguir, a mi hermana y para todos aquellos que depositaron su confianza en mí y me apoyaron en mi camino hacia la suficiencia profesional, les dedico este trabajo. Su amor, aliento y motivación me han ayudado a llegar hasta aquí y por ello estoy muy agradecido. También quiero dedicárselo a mis docentes y mentores, cuya guía y enseñanza han sido fundamentales para mi formación y crecimiento profesional. Espero que este trabajo les haga sentir orgullosos y que sea una muestra de mi compromiso y dedicación a mi carrera.

## **AGRADECIMIENTOS**

En primer lugar agradecer a Dios por ser quien me guía día a día, por brindarme salud y la fortaleza necesaria en los momentos difíciles que me ha tocado enfrentar en los diferentes momentos de mi vida.

A Nilton Quispe Mayuri y Epifania Garayar Casavilca, mis amados padres, por su sacrificio, por ser mi soporte e inspiración de seguir adelante y no rendirme en el camino hasta conseguir mis objetivos y por regalarme su mejor herencia, la educación y valores. A mi hermana y a las personas que confiaron en mí y me acompañaron en el proceso.

## Índice de contenidos

Dedicatoria	II
Agradecimientos	III
Índice	IV
- Índice de contenidos	
- Índice de tablas	V
- Índice de figuras	VI
Resumen	VII
Abstract	VIII
<b>CUERPO DEL INFORME</b>	
INTRODUCCION	1
Antecedentes	2
Fundamentación teórica	7
SCRUM	7
Jira	10
TestLink	11
Planteamiento del problema	14
Objetivos	15
Alcance	15
Limitaciones	16
Justificación	16
Capítulo I: INFORMACION DE LA INSTITUCION DONDE SE DESARROLLO LA EXPERIENCIA	17
Capítulo II: TRAYECTORIA PROFESIONAL	19
Capítulo III: APLICACION PROFESIONAL	20
Situación problemática	20
Proyecto desarrollado	22
Herramientas utilizadas	22
Resultados	44
Capítulo IV: APORTES A LA INSTITUCION	46
CONCLUSIONES	47
RECOMENDACIONES	48
REFERENCIAS BIBLIOGRÁFICAS	49
ANEXOS	51
Anexo 01: Equipo SCRUM y QA	51
Anexo 02: Contrato experiencia profesional	52

## INDICE DE TABLAS

Tabla 1 Tipos de pruebas con TESTLINK	12
Tabla 2 Alcance de los requerimientos	16

## INDICE DE FIGURAS

Figura 1. Servicios ofrecidos por NTT DATA	1
Figura 2. Sectores a los que atiende	1
Figura 3. Empresas Grupo Intercorp	1
Figura 4. Metodología Agile SCRUM	8
Figura 5. Seguimiento de las tareas	10
Figura 6. Estructura de Jira	10
Figura 7. Funcionalidades del aplicativo	14
Figura 8. Monitoreo de la satisfacción de los clientes	15
Figura 9. Sucursales de NTT DATA	17
Figura 10. Corporación NTT DATA	18
Figura 11. Estructura orgánica de NTT DATA	18
Figura 12. Problemática detectada	21
Figura 13. Plan de Entrega del Q2	22
Figura 14. Planeamiento del Q2	23
Figura 15. Daily del Q2	23
Figura 16. Refinamiento del Q2	23
Figura 17 de la Retrospectiva Q2	24
Figura 18. Review del Sprint	24
Figura 19. Mural usado para refinamiento y planning de los del Sprint	24
Figura 20. Mural usado para la retrospectiva de cada Sprint	25
Figura 21. Especificación de la HU de un usuario del Tunki	25
Figura 22. Gestión de las tareas finalizadas en Jira	26
Figura 23. Issues encontrados durante las pruebas dentro de una HU de Tunki	26
Figura 24. Historia de usuario para QR interoperable usando figma	27
Figura 25. Historia de usuario para la recarga del celular usando Invision	27
Figura 26. Desarrollo con la herramienta Invision	28
Figura 27. Rediseño con la herramienta Invision	28
Figura 28. Rediseño sin Rebranding con la herramienta figma	29
Figura 29. Rediseño con Rebranding con la herramienta figma	29
Figura 30. Desarrollo sin Rebranding con la herramienta figma	30

Figura 31. Desarrollo con Rebranding con la herramienta figma	30
Figura 32. Testeo con Postman con el método Get	31
Figura 33. Automatizador de los flujos en las aplicaciones	32
Figura 34. Mapeo de la HU y verificar requisitos	33
Figura 35. Flujo redactado en el lenguaje Gherkin	34
Figura 36. Como ejecuta las descripciones de la Fig. 34	35
Figura 37. Herramienta para mantenimiento del código	36
Figura 38. Interfaz de comunicación con el servidor	37
Figura 39. Interfaz de control y monitoreo de los errores que se puedan presentar	37
Figura 40. Monitoreo de los tiempos de respuestas y otros parámetros	38
Figura 41. Exploración de los puntos finales para las pruebas de la HUs	39
Figura 42. Comunicación mediante la VPN para la realización de las pruebas	39
Figura 43. Seguridad de la conexión	40
Figura 44. Personalización del servidor para Android	41
Figura 45. Utilización del gestor de datos Postgres	42
Figura 46. Creación y Acuse de tarjeta de crédito	42
Figura 47. Consultas mediante SPU	43
Figura 48. Monitoreo y ejecución mediante Testlink	44
Figura 49. Interfaces de bienvenida	44
Figura 50. Interfaces seguridad de acceso	45
Figura 51. Evidencias de resultados de los sprint	45
Figura 52. Equipo de desarrollo SCRUM, dev (front end y back end), QA y los arquitectos de software	51
Figura 53. Reuniones diarias del equipo	51

## **RESUMEN**

El presente trabajo de suficiencia profesional desarrollada para la empresa financiera Interbank en la mejora de su aplicativo Tunki, por medio de nuestra empresa prestadora de servicios tecnológicos en el mundo NTT DATA, nace como una necesidad de prestación de mejores servicios de este aplicativo, debido a los comentarios negativos sobre el mismo. Por lo cual es objetivo es mejorar la experiencia de los usuarios con el aplicativo Tunki, la metodología para lograr este objetivo fue SCRUM para el desarrollo y las técnicas de QA para asegurar la calidad del desarrollo y cumplimiento de los requerimientos establecidos. Los resultados se reflejan en nuevos comentarios relacionados con la mejora de la utilidad del aplicativo Tunki. Se concluye que se ha logrado el objetivo de mejorar el aplicativo reflejado en la satisfacción de los clientes.

Palabras claves: QA, SCRUM, Aplicativo Tunki

## **ABSTRACT**

The present work of professional sufficiency developed for the financial company Interbank in the improvement of its Tunki application, through our technology service provider in the world NTT DATA, was born as a need to provide better services of this application, due to the negative comments about it. Therefore, the objective is to improve the user experience with the Tunki application, the methodology to achieve this goal was SCRUM for the development and QA techniques to ensure the quality of development and compliance with the established requirements. The results are reflected in new comments related to the improvement of the usefulness of the Tunki application. It is concluded that the objective of improving the application has been achieved as reflected in customer satisfaction.

Keywords: QA, SCRUM, Tunki application.

## INTRODUCCION

Las necesidades de las organizaciones en mejorar sus procesos de negocio por medio de la tecnología informática, día a día crece, en todos los sectores de la industria, siendo el tratamiento de esta información muy delicada, la corporación NTT DATA es una empresa internacional dedicada a ofrecer diversos servicios en tecnología informática, como se puede comprobar en su portal web.

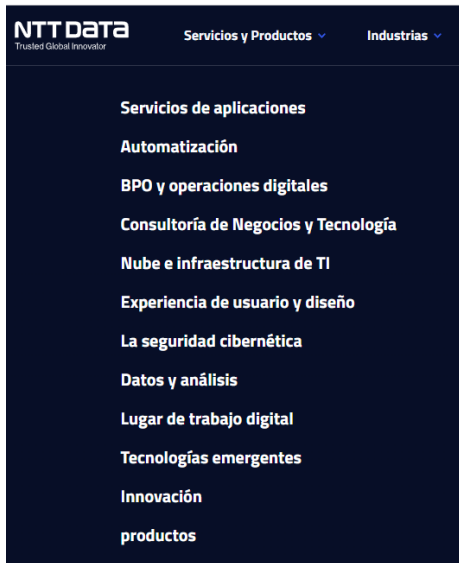


Figura 1. Servicios ofrecidos por NTT DATA

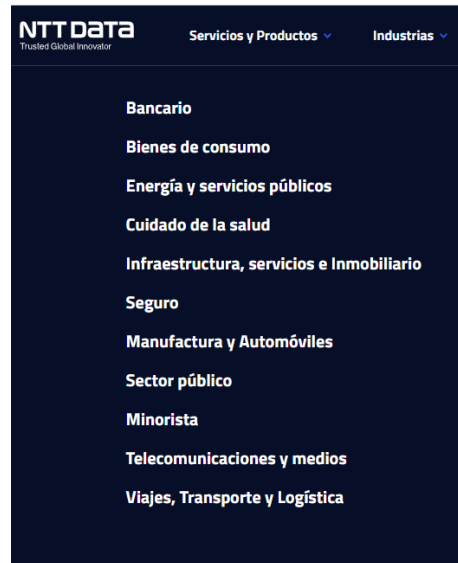


Figura 2. Sectores a los que atiende

Como se desprende de la Figura 2, uno de sus principales sectores de atención es el sector bancario. La empresa atiende a este sector en los diversos países en donde tiene presencia, como es el caso de nuestro país.

En el Perú el grupo Intercorp, es una corporación que tiene diversas empresas en su organización como se muestra en la Figura3.

### Empresas

- Interbank. Artículo principal: Interbank. ...
- Intercorp Retail. ...
- Casa Andina. ...
- Supermercados Peruanos S.A. ...
- Tiendas Peruanas. ...
- Homecenters Peruanos. ...
- Financiera Oh! ...
- Interproperties.

Figura 3. Empresas Grupo Intercorp

Cómo se puede apreciar en la Figura 3, dentro de las empresas del grupo Intercorp se encuentra en el sector bancario la empresa Interbank su principal empresa como ahí se detalla; empresa a la cual la corporación NTT DATA, presta sus servicios.

El tema se eligió considerando mis actividades en la empresa Everis del grupo NTT DATA en el servicio prestado al banco Interbank, que es cliente del grupo NTT DATA. Interbank tuvo una necesidad de mejorar su aplicativo **Tunki** con la finalidad de ofrecer una mejor experiencia a sus clientes en la realización de sus transacciones financieras, ante esta necesidad se realiza el presente proyecto que consiste en asegurar la calidad de la mejora del aplicativo con un enfoque en la calidad del producto por medio de herramientas de QA.

## 1. Antecedentes

La constante transformación que están pasando las organizaciones, igualmente sucede con el banco líder de Colombia Bancolombia cuya presencia en el mercado data de más de 146 años, y que para lograr satisfacer las necesidades de sus clientes es que presentará la automatización de su aplicativo Adminfo Vsmart, aplicativo orientado a solucionar la gestión de diversos procesos como la cobranza, su cartera administrativa, y diversas carteras de sus clientes como los bienes recibidos en parte de pago. Para lograr esto el banco realiza un análisis y estudio de estas carteras para ver los contenidos en el área de tecnología y la metodología para el proceso de asegurar la calidad por medio de patrones de diseño, con ello se realizó la planificación de estas pruebas por medio del procedimiento de ruta crítica y el módulo de mantenimiento, para establecer esta ruta se emplea como técnica un diagrama de flujo por medio de la plataforma de Bizagi, y con esta flujograma con el software Gherkin se desarrolla entre 1 y 3 transacciones por perfil, vale decir lo que debe hacer cada perfil. Se realiza un levantamiento de los riesgos para iniciar con el plan de pruebas, a fin de lograr la mejorar la calidad del aplicativo en desarrollo. Avanzado el proyecto y con las ejecuciones respectivas en varias etapas y perfiles, se configura el Release Management (RM) y el tablero Hygieia con las métricas establecidas por el banco, con el objetivo de incrementar los porcentajes de medición encima del 80% al finalizar la ejecución de todas la pruebas en el RM [1].

En el proyecto donde las aplicaciones en las que se utilizan los datos de manera intensiva son críticas y las más demandadas para la mayoría de las prácticas habituales, como son los sistemas de recomendación. Los medios sociales son medios muy importantes para el análisis de las noticias falsas o fake news, pero lamentablemente para mostrar este tipo de soluciones en tiempo real, las pruebas de software y asegurar la calidad son de suma importancia para entender el funcionamiento de la aplicación. La estructura de las 4V para los macrodatos por su nivel de complejidad y retos que son necesarios abordarlos en el desarrollo de estas aplicaciones en tiempo real. Las pruebas realizadas para aplicaciones de BigData demuestran

ser eficientes y de modelan y ejecutan planes de pruebas, pero hay que considerar todas las V que forman el ecosistema del BigData. Este manual presenta de forma explícita los aspectos teóricos y prácticos de las pruebas de calidad para soluciones en enormes cantidades de datos. se presentan y discuten diversas pruebas para este tipo de datos por medio de herramientas como Hadoop, Flink, Spark, Zookeeper, Hive, NoSQL, Flume y muchas otras más orientadas al tratamiento del BigData. La unidad de pruebas realizadas con MapReduce, las pruebas de Kafka y las pruebas y biblioteca de pruebas adicionales se han integrado con casos de pruebas y pruebas de caso impulsado en el desarrollo de la aplicación [2].

El aumento en la demanda de los sistemas de información es una necesidad en el tiempo y el aseguramiento de la calidad de estos sistemas, en el presente trabajo se tuvo como objetivo proponer el empleo de metodologías ágiles para asegurar esta calidad del software aplicada en una institución del estado, con la finalidad de obtener resultados rápidos que permitan una mejora en los requerimientos de la institución estatal. La metodología utilizada corresponde a una investigación del tipo cualitativo, básica con una investigación con un diseño de investigación acción. Para lograr el objetivo se preparó un guía de entrevista semiestructurada y que fue validada mediante juicio de expertos en metodologías ágiles, adicionalmente se utilizó las técnicas de la observación y el análisis documental para el recojo de datos. por medio de la triangulación se pudo contrastar el enfoque por medio de estos datos y sacar las conclusiones relacionadas con los objetivos planteando. El trabajo concluye que para la utilización de las metodologías ágiles como SCRUM o KAMBA se requiere contar con ciertos factores de importancia como la capacitación del personal en el manejo de los métodos y principios, el establecimiento de los roles es de vital importancia para el desarrollo del software, cuyo equipo deben ser auto organizados y con espacios co-ubicados en las que se fomenta la colaboración y comunicación entre todos [3].

En otro escenario, la responsabilidad que tienen las instituciones formadoras de ingenieros de software tiene la obligación de orientar sus cursos en el fortalecimiento de habilidades y destrezas para el proceso de desarrollo e implementación de software, y en la que prime la calidad como un aspecto principal para que se desarrollen productos que satisfagan los requerimientos de los usuarios. En este trabajo de investigación se muestran las mejores prácticas del aseguramiento de la calidad en el desarrollo de proyectos de automatización de procesos académicos en el marco de la universidad, evaluando los errores que se presenten en los equipos de desarrollo, encontrando las oportunidades de mejoras en el proceso de desarrollo de aplicaciones, al emplear las buenas prácticas de calidad. SCRUM fue la metodología utilizada como buenas prácticas de desarrollo. Se concluye que se observó la madurez en los equipos de trabajo en la que se ve mejoradas las habilidades y destrezas [4].

En el presente trabajo cuyo objetivo es determinar la manera en que el aseguramiento de la calidad del desarrollo de un software influye en los proyectos desarrollados por la empresa 3M. Para lograr esto el equipo de pruebas implementadas fue conducido por la empresa QSYSTEM, la que tuvo como misión encontrar las limitaciones en las aplicaciones que emplea la empresa 3M y que fueron probadas en los proyectos que se encuentran en ejecución. Previo al estudio no se contaba con un área destinada a las pruebas que encuentre las fallas que eviten en el tiempo diversos errores que se dan cuando estos ya se encuentran en producción, ya que esto genera mayores gastos. La investigación de tipo aplicativo tuvo un enfoque cuantitativo, con un diseño preexperimental, y corte longitudinal, para lo cual se estableció una muestra de 15 colaboradores de la empresa. Se empleó como técnica de validación de los instrumentos por medio del Alfa de Cronbach en el pretest = 0.832 y en el postest = 0.706 con un nivel de significación de 0.000 que respalda la hipótesis planteada. El trabajo concluye que si existe una diferencia significativa en los proyectos que desarrolla la empresa 3M en el aseguramiento de la calidad antes y después de su puesta en aplicación, lo que impactó positivamente, reduciendo notablemente los errores del aplicativo cuando ya se encuentran en producción, además de que muchas de estas aplicaciones no se encontraron errores [5].

En este trabajo de fin de máster, las pruebas para asegurar la calidad de un software o aplicación cumplen un rol importante en cualquier proyecto de desarrollo de software. Por lo que el objetivo es fundamentalmente asegurar la calidad del producto final, por medio de un proceso que incluye las fases de diseño, implementación, pruebas y despliegue e integración. Los profesionales destinados a asegurar la calidad que trabajan en cada una de estas etapas del ciclo de vida en el desarrollo de un software emplean para ello metodologías ágiles, en la que se construyen pequeños fragmentos de software por medio de un proceso iterativo e incremental, asegurando la calidad de cada entrega. En la actualidad se están recurriendo a la Inteligencia artificial en el campo del aprendizaje automático, apoyan de forma natural estos procesos iterativos para el aprendizaje continuo. Con estas nuevas herramientas se requieren asegura la calidad de los datos que entran al modelo, y se evalúa la salida según lo esperado por el modelo entrenado previamente y validado por medio de métricas de estos modelos. En base a esto se hace necesario la integración de diversos modelos, algoritmos, heurísticas y las etapas propias de un modelo basado en inteligencia artificial en las que se incluyen las pruebas y el control de la calidad del producto. Se requiere que estos procesos sean integrados en cada una de las etapas como también en la etapa del mantenimiento para una mejora continua para asegurar la calidad de los cambios del aplicativo. En la investigación se incluye la revisión de los procesos y las herramientas para asegurar la calidad del software y su integración con el aprendizaje automático de la inteligencia artificial. Igualmente, se evalúa nuevas metodologías

como Data Centric y MLOps en el aseguramiento de la calidad para implementación compleja basado en la IA. Finalmente, se plantea la utilización de estas estrategias y criterios para asegurar la calidad en un caso de uso en el área del procesamiento automático de documentos [6].

Los proyectos de desarrollo de software en las que participan colaborativamente instituciones y personas con diversas características de conocimiento y formación que los orienten al desarrollo de los productos digitales de calidad. En este estudio es suma importancia ya que el aporte de la experiencia de cada una de estas personas, permiten alinear expectativas siempre que se incorporen los puntos de aseguramiento de la calidad en las fases iniciales del desarrollo de proyectos de software. El uso de las metodologías ágiles son ya una costumbre en los equipos de desarrollo con la incorporación de equipos multidisciplinares, para lo cual se cuenta con tres grandes etapas: Inception o etapa de inicio, Planning o etapa de planeación y Implementation o etapa de implementación. Lamentablemente los procesos de control de la calidad del producto se realizan en la etapa de implementación, dejando las dos primeras fuera de este proceso. El presente estudio muestra de una forma práctica y respaldada por dos proyectos reales: iMedea y Meet2Care, para integrar los procesos de aseguramiento de la calidad desde las etapas iniciales del desarrollo ágil, incluyendo a los involucrados ya que es de vital importancia para lograr el éxito del proyecto. Para los proyectos mencionados se presenta una propuesta, la misma que ha tenido el éxito en su implementación, con las herramientas de calidad en las que se usan desde las etapas iniciales e involucrando a todos los involucrados. Los proyectos tomados como casos fueron orientados en el sector [7].

En el estudio de investigación, la calidad de software se refiere a los diversos atributos que lo hacen atractivo para el que lo adquiere y para el que lo va a operar. Siendo estos atributos intangibles, con la finalidad de su Inter operatividad, reutilización, uso, etc. Desde mucho tiempo atrás se ha ido incorporando el término de calidad en un gran número de características internas y externas, considerando que se habla de algo más que el simple costo de producto. En tal sentido se ha de suma importancia cumplir con ciertos estándares mínimos para asegurar la calidad y es aquí donde aparecen los criterios de las métricas para medir esa calidad. Estos instrumentos que nos permiten medir el uso de todos los atributos de interés en un software. El proyecto tuvo como objetivo identificar los aspectos teóricos sobre la calidad del software y el empleo de estas métricas. La metodología utilizada en la investigación se basó en la búsqueda y revisión bibliográfica, utilizando descriptores: software, calidad del software. El resultado arrojó 15 fuentes bibliográficas por medio de Google académico y la base de datos Scopus. Estas fuentes seleccionadas cubrían nuestro interés en la calidad del software con no menos de 5 años de antigüedad. El estudio concluye que los modelos de calidad de software

en gran medida han sido desarrollados por las empresas desarrolladoras de software, a fin de medir los atributos que posee esta. Los atributos fueron agrupados en criterios que pueden ser medidos, los cuales pueden igualmente ser verificados de forma empírica y teórica [8].

Las herramientas y técnicas para asegurar la calidad de un software incluyen diversas pruebas, inspecciones y revisiones por parte del equipo de desarrollo, pruebas habituales en los proyectos de desarrollo. En esta investigación se analiza la aplicación y eficacia de estas técnicas con las herramientas habitualmente disponibles y que son fáciles de utilizar por estudiantes de sistemas, al mismo tiempo se utilizan casos concretos de software para hacer ver las mejores prácticas en el trabajo y evaluar los datos resultantes con la finalidad de poder facilitar la toma de decisiones en proyectos que conlleven a mejorar los resultados en la calidad del producto [9].

En la actualidad, la necesidad de los encargados de probar la cada vez más creciente cantidad de software en periodos de tiempo más cortos. Esta situación hace casi imposible utilizar suites de pruebas completas en cada cambio realizado al software. Lo que a su vez se ha convertido en una práctica por eficiente, asegurar la calidad del software por medio de pruebas manuales no son posibles cubrirlas todas. Por lo que en el presente estudio se propone el Change-DrivenTesting y Test-Gap Analysis, el mismo que emplea un análisis del impacto de las pruebas con la finalidad de encontrar de forma automatizada las pruebas respectivas para cualquier tipo de modificación de código y agruparlas de manera que incremente la posibilidad de detectar errores en las fases iniciales del desarrollo. Las pruebas automatizadas, son más eficientes, ya que estas son posible de detectar más del 90% de los posibles errores en un tiempo de solo el 2% en las pruebas. Estas dos herramientas permiten la identificación de algunas lagunas en las pruebas, como los cambios en el código. Lo que a su vez nos lleva a tomar decisiones más acertadas e informadas sobre en donde utilizar los escasos recursos que normalmente se poseen, logrando mejorar la eficacia de estas pruebas, porque además nos informa sobre los faltantes por medio de pruebas de regresión [10].

Desde la aparición de la tecnología Grid en el campo de la computación distribuida, hasta los ya conocidos modelos de la computación en la nube, se hace necesario de nuevas herramientas y servicios que permitan atender la creciente demanda de necesidades en la comunidad científica. Estas se han incluido en Europa por medio de proyectos de desarrollo de software orientados a la gestión, creación y operación de e-infraestructuras. En la investigación se muestran los avances más importantes en este campo de la calidad del software desarrollados en uno de los proyectos denominado INDIGO-DataCloud, en este ámbito se tiene algunos retos y limitaciones que se han encontrado en la vida un proyecto, y como fueron abordados y

superados en algunos casos parcial y en otros totalmente. En los últimos 15 años se han establecido muchas iniciativas en el desarrollo de software en la comunidad europea, con la finalidad de sostener infraestructuras electrónicas en el campo de la investigación distribuida, y sostenidas por los avances en el campo de la ingeniería de software, lo que contribuyó si duda en la calidad del producto, la fiabilidad de los softwares desarrollados y como tal la estabilidad en las operaciones de estas e-infraestructuras en Europa. El proyecto INDIGO-DatCloud es una muestra de esto, en relación con proyectos anteriores, las utilización de prácticas para asegura la calidad del software siempre han estado presente desde las fases iniciales del ciclo de vida de un software [11].

La calidad de un producto siempre va a ser el foco de atención en la mejora y evaluación de estos, a pesar de que muchas de las investigaciones que se han llevado a cabo en el campo de las mejoras de la calidad interna. Si embargo se ha prestado menos atención desde la perspectiva de los usuarios para mejorar la calidad del software. Sin duda los usuarios son los llamados a querer mejorar la calidad y usabilidad de un software. Sin embargo, hay que tener presente que la calidad de un software en realidad obedece a satisfacer las necesidades o requerimiento de los usuarios de ese software. En el presente estudio, se presentan algunas métricas y usos para asegurar la calidad del software, luego se ha recopilado la mayoría de estas métricas obtenidas de la literatura las mismas que se han clasificado en un grupo de 27 métricas. Con este grupo de métricas se realizó una encuesta con los estudiantes, sobre las perspectivas que debe tener un usuario para clasificarlas en base a la importancia en la calidad del software. Con las respuestas se pudo construir un nuevo modelo de aseguramiento de la calidad denominado calidad desde la perspectiva del usuario [12].

Los profesionales que se encuentran en la línea de producto de software (SPL) realizan diversos esfuerzos para asegurar que, por medio de una gran cantidad de posibles configuraciones, los productos puedan ser aceptados y adaptados a los requerimientos de los clientes. Sin embargo, esta perspectiva genera una gran cantidad de espacios de configuración casi imposibles de poderlas cumplir, y requiere de una exhaustiva revisión y análisis. Por ello ante esta limitación y con el enorme potencial que viene dañado el aprendizaje automático se pueden con esta herramienta tener una aproximación de productos que sean aceptados por los clientes, para ello se parte de una muestra de configuraciones, y con el aprendizaje automático se pueden obtener una línea de productos de software por medio de las características aprendidas y reducir a priori productos no deseados. El trabajo se emplean técnicas de Machine Learning (ML) para crear configuraciones adversarias que engañan a los clasificadores de ML e ubican clasificaciones incorrectas de productos (videos) que ha sido creados desde un video institucional. Los ataques realizados generan una tasa de clasificación con errores del 100%

con una reducción de la precisión del algoritmo del 5%. Se discutió las discrepancias de los resultados en el aseguramiento de la calidad de los SPL [13].

## 2. Fundamentación teórica

2.1 SCRUM: Scrum trabaja de una a cuatro semanas en un sprint en el que su producto es definido, diseñado, creado y probado por usted. Y luego muestra su producto a las partes interesadas para ver si debemos ajustarlo o hacer algo diferente. Y luego, y se repite el ciclo una y otra vez. De esta manera, crea su producto gradualmente después de cada sprint. Y al final, obtiene todo su producto. En un método en cascada, durante la fase de requisitos, recibe un documento de requisitos al final. Y luego viene la fase de diseño, donde obtiene documentos del diseño de alto nivel o de diseño de bajo nivel, tiene más documentos de diseño. Luego la implementación, verificación y entonces hace el lanzamiento y finalmente, obtiene su producto terminado. Como puede ver, en el método en cascada, tiene un gran lote y luego su producto final. Mientras que, en Scrum, cada dos, o entre 1 y 4 semanas, tiene un producto terminado y le sigue añadiendo elementos hasta que obtenga su producto final. Ahora que tenemos una idea básica de Scrum, veamos de cerca qué sucede en cada uno de estos sprints de una a cuatro semanas. Por ejemplo, una compañía quiere crear un portal de empleo, digamos como monster.com, y quieren usar un marco de Scrum.

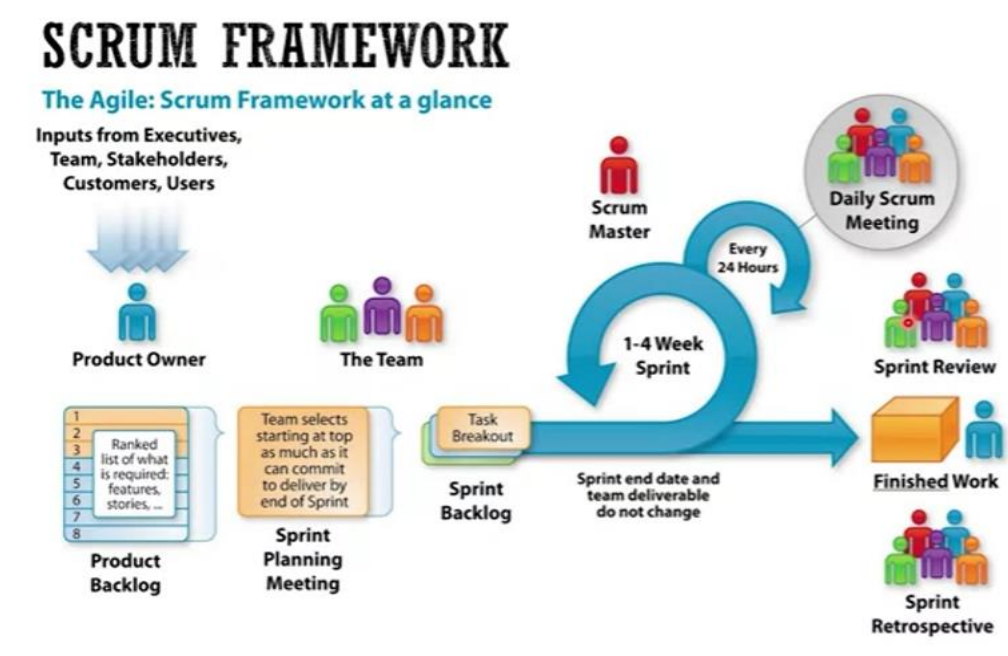


Figura 4. Metodología Agile SCRUM

En la Figura 4, se puede apreciar que existen tres roles que definir en Scrum. Primero el propietario del producto (Product Owner) que define qué se debe hacer y en qué orden.

Y luego hay un facilitador que ayuda al equipo a mantenerse fiel a los valores y principios de Scrum (Scrum Master). También notará que el facilitador asiste en la mayoría de las reuniones del equipo. Y si hay algún obstáculo, él será el que conducirá a la resolución de algunos de estos obstáculos. Y luego, por supuesto, está el equipo (The Team), que es autónomo y hace la mayor parte del desarrollo del software, incluye a los desarrolladores y probadores.

Procedimiento: El Product Owner hablará con ejecutivos, el equipo, interesados, clientes, usuarios e intentará definir qué es exactamente lo que necesita crear. Eso creará algo llamado lista de tareas, que es básicamente una lista de historias de usuarios que se priorizan y definen lo que debe hacerse. La lista de tareas es muy diferente a un típico documento de requisitos. Como verá, es a un nivel muy alto, y puede cambiar con el tiempo. Un nivel muy alto. ejemplo, el primer elemento puede decir "Publicar un empleo". Un administrador puede publicar un empleo o un solicitante puede solicitarlo. A ese nivel, la lista de tareas está definida. Cuando el equipo esté listo para el sprint, hace una reunión de planificación de sprint, donde todo el equipo se reúne y eligen las historias principales para incluir en el sprint existente. El propietario del producto revisa esas historias con el equipo y ayuda a responder preguntas o aclarar cualquier duda. Luego, todo el equipo se reúne de nuevo y trabaja a partir de las historias, respondiendo a qué se necesita para crear el software. En este caso, pueden decir: diseñar la base de datos, llenar algunos datos, y testear la base de datos. Necesito crear una interfaz, etcétera, etcétera. Una vez que el equipo haya visto y asignado las historias, saben que pueden terminar en el próximo sprint o el sprint existente, se dedican a esas historias y dicen: "Terminaremos en este sprint". Y luego comienza su ejecución.

Y luego comienza su ejecución. Eso inicia un sprint en el que todos trabajan para implementar el software. Y durante ese sprint, todo el equipo se reúne en una reunión informal diaria, hablan sobre lo que hicieron ayer, qué van a hacer hoy, y si hay algún obstáculo. En este caso, alguien de la base de datos puede decir: "Diseñé las tablas ayer en la base de datos. Y hoy, voy a completarlas con unas pruebas de datos, si hay alguien que pueda ayudarme, sería increíble". Y también puede decir: "No puedo acceder a ciertas cosas en la base de datos, a menos que tenga ese permiso, no podré terminar mi trabajo". Entonces, alguien podría ayudarlo, como ¿con quién me comunico, ¿qué hago para obtener acceso a eso? Así es el trabajo el resto de los días en el sprint, pero al final, obtendrá su producto terminado.

Pasos en la planificación de un Sprint:

1. Determinar la capacidad de Sprint
2. Revisar el objetivo del sprint
3. Revisión de las Historias de Usuario potenciales

4. Asignar tareas y discutir el diseño por parte del equipo
5. Refinar las Historias de Usuario, sino se cumple el objetivo
6. Compromiso de terminar las Historias de Usuario
7. Poner en el muro de tareas las Historias de Usuario y comienzan a ejecutarse

## TASK BOARD

Story	To Do	In Progress	Done
Story A		Task	Task
Story B	Task	Task	Task
Story C		Task	Task

Figura 5. Seguimiento de las tareas

En la Figura 5, se muestra como realizar el seguimiento de las tareas, para lo cual se emplea un Tablero de tareas para ir realizando el control en tres fases: Por hacer (To Do), en ejecución (In Progress) y culminadas o listas (Done) [14].

2.2 Jira: “Jira es una aplicación web que con el tiempo se ha convertido en el estándar del mercado en las áreas de gestión de proyectos, gestión de tareas y gestión de errores” [15]. Es un software en la nube para la gestión de proyectos, con Jira se consigue una planificación más inteligente y una comunicación mejorada con las hojas de ruta de Jira Software Mantén tus equipos y tu organización sincronizados con software de hojas de ruta integrado para que publicar sea un proceso más predecible y mantener el enfoque en el objetivo a fin de alcanzar las metas como se muestra en la Figura 6.

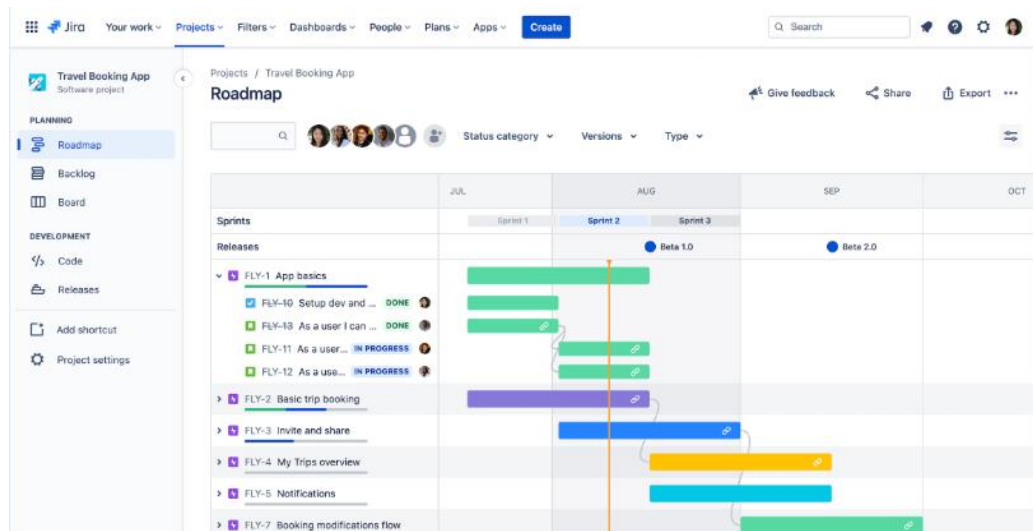


Figura 6. Estructura de Jira

En la Figura 6, se puede apreciar que Jira es una excelente herramienta para el desarrollo de proyectos mediante la metodología agile, en el panel izquierdo se puede apreciar que se puede registrar el backlog, que es donde se tienen todas las historias de usuario que conforman el producto backlog. Y de donde se van seleccionando los Sprints para desarrollar en cada ciclo de desarrollo.

En el panel central se tiene para hacer el seguimiento del sprint que este en desarrollo e ir cambiando su estado (In progress, Done).

2.3 TestLink: TestLink es una herramienta de uso libre para la gestión de casos de pruebas que les permite a los miembros de un equipo realizar un seguimiento de diversas pruebas de manera dinámica en proyectos de desarrollo de software y manejar la trazabilidad con los requerimientos [16].

En la Tabla I, siguiente se proporciona una visión general de cómo las diferentes pruebas se relacionan con las funcionalidades de gestión de pruebas que ofrece TestLink. Ten en cuenta que TestLink es una herramienta flexible y personalizable, por lo que también puedes adaptarla para incluir otros tipos de pruebas específicos de tu proyecto.

Tabla 1  
Tipos de pruebas con TESTLINK

<b>Tipo de Prueba</b>	<b>Descripción</b>	<b>Funcionalidades de TestLink</b>
<b>Pruebas Unitarias</b>	Verificar el correcto funcionamiento de componentes individuales del software.	Creación de casos de prueba detallados, asignación de casos de prueba a versiones o ciclos de prueba, ejecución y seguimiento de resultados.
<b>Pruebas de Integración</b>	Verificar la interacción y la comunicación adecuada entre los diferentes componentes del sistema.	Creación de suites de prueba que agrupen casos de prueba relacionados, seguimiento de resultados de pruebas, gestión de problemas y defectos.
<b>Pruebas de Sistema</b>	Verificar si el sistema en su conjunto cumple con los requisitos y funcionalidades especificados.	Creación de casos de prueba con precondiciones y postcondiciones, seguimiento de cobertura de pruebas, generación de informes de pruebas.
<b>Pruebas de Aceptación</b>	Determinar si el sistema cumple con los criterios de aceptación definidos por los usuarios o los stakeholders.	Creación de casos de prueba basados en requisitos y criterios de aceptación, gestión de versiones de casos de prueba, seguimiento de resultados.
<b>Pruebas de Regresión</b>	Garantizar que las modificaciones o actualizaciones realizadas en el software no hayan introducido nuevos problemas o hayan afectado negativamente funcionalidades existentes.	Gestión de versiones y ciclos de prueba, ejecución de pruebas previas y posteriores a cambios, seguimiento de resultados y comparación de versiones.
<b>Pruebas de Rendimiento</b>	Evaluar el rendimiento, la capacidad y la estabilidad del sistema en situaciones de carga y estrés.	Creación de casos de prueba específicos de rendimiento, ejecución de pruebas de carga y estrés, seguimiento de resultados y generación de informes.

<b>Pruebas de Seguridad</b>	Identificar y evaluar posibles vulnerabilidades y riesgos de seguridad en el sistema.	Creación de casos de prueba específicos de seguridad, ejecución de pruebas de seguridad, gestión de problemas y defectos relacionados con seguridad.
-----------------------------	---	--

### 3. Planteamiento del problema

Como toda institución está sujeta a los cambios de entorno y del mercado y constantemente deben adecuarse a este entorno cambiante, lo que crea una necesidad continua y por lo cual hay que mejorar el servicio al cliente, en tal sentido entre las necesidades de mejora de su aplicativo **tunki** del Interbank se destacan:

1. Rediseño: siempre buscan hacerlo más amigable
2. Promociones que dan
3. Seguridad: ya que tiene un área de ello, donde encuentran vulnerabilidades, entonces se dan soluciones y en base a ello hacemos pruebas
4. Nuevos requerimientos que surgen
5. Correcciones de incidencias que se obtienen revisando comentarios que dejan en la tienda (play y app store) para ambos sistemas operativos Android y Iphone.



Figura 7. Funcionalidades del aplicativo

En la Figura 7, se puede apreciar alguna de las funcionalidades que tiene el aplicativo, y en la que se han diseñado y desarrollado las mejoras para poder satisfacer las necesidades de los clientes del banco Interbank, aplicando técnicas de QA.



Figura 8. Monitoreo de la satisfacción de los clientes

En la Figura 8, los comentarios negativos sobre la funcionalidad del aplicativo, que sin embargo se ha ido revertiendo con los cambios y mejoras como se precisa en el comentario de Guillermo Tielca y Miguel Huamán. Esta situación es de mejora continua según el monitoreo de los comentarios de los clientes.

#### 4. Objetivos

Objetivo general: Satisfacer las necesidades de los usuarios del aplicativo tunki.

Objetivos específicos:

- ✓ Mejorar continuamente las funcionalidades del aplicativo
- ✓ Mejorar la experiencia de los clientes rediseñando el aplicativo haciéndolo más amigable en base a los principios de usabilidad.
- ✓ Desarrollar los nuevos requerimientos que se generan

#### 5. Alcance

El alcance del proyecto se limita a las mejoras que se deben realizar al aplicativo tunki, en aquellas necesidades relacionadas con la experiencia de los usuarios y que a la fecha de la petición del cliente no están siendo cubiertas, por lo que el alcance consiste en asegurar la calidad de los requerimientos de usabilidad y erros presentados del aplicativo, para asegurar la satisfacción de los clientes.

El alcance se detalla en la Tabla 2 de los requerimientos, los que han sido clasificados en las Historias de Usuario (HU) para cada Sprint a desarrollar.

Tabla 2  
Alcance de los requerimientos

<b>Sprint 1</b>	<b>Requerimiento</b>
<b>HU TKGZ-9</b>	Activación de preafiliados
<b>HU TKGZ-21</b>	Modificación TAG
<b>HU TKGZ-331</b>	Merge intero + izi
<b>HU TKGZ-1327</b>	Afiliación a izi
<b>HU TKGZ-312</b>	Rediseño de flujos restantes
<b>Sprint 2</b>	
<b>HU TKGZ-42</b>	Mejora en ver QR
<b>HU TKGZ-331</b>	Merge intero + izi
<b>Sprint 3</b>	
<b>HU TKGZ-311</b>	Envío de constancias
<b>HU TKGZ-21</b>	Modificación TAG
<b>HU TKGZ-338</b>	Validación online registro
<b>Sprint 4</b>	
<b>HU TKGZ-333</b>	Deeplink push notification
<b>HU TNT2- 1190</b>	Actualizar correo
<b>HU TKGZ-311</b>	envío de constancia
<b>HU TKGZ-612</b>	Ajuste de activación de preafiliados
<b>Sprint 5</b>	
<b>HU TNT2- 1190</b>	Actualizar correo
<b>HU TKGZ-327</b>	Actualizar número de celular
<b>Sprint 6</b>	
<b>HU TKGZ-617</b>	Bloqueo de intentos

#### 6. Limitaciones

En el proyecto no se han presentado limitaciones para realizar el trabajo de atención al cliente, para el desarrollo del proyecto se ha tenido todas las facilidades de la corporación Intercorp y concordancia con la corporación NTT DATA por estar en sus prioridades la mejor atención a clientes como Interbank.

#### 7. Justificación

El proyecto está justificado por la necesidad del cliente, para este caso el Interbank, porque la mejora del aplicativo **tunki** va a beneficiar a una mayor cantidad de clientes que usan este aplicativo. Además, se justifica porque la empresa NTT DATA afianza su compromiso con sus clientes corporativos, mejora su presencia en el mercado.

## Capítulo I: INFORMACION DE LA INSTITUCION DONDE SE DESARROLLO LA EXPERIENCIA.

NTT DATA, es una corporación que presta servicios de TI a nivel mundial, cuya sede principal de encuentra ubicada en la ciudad de Tokio Japón, la misma se encuentra ubicada en el 6º lugar del ranking de compañías proveedoras de servicios de TI en todo el mundo, la corporación NTT DATA tiene presencia en más de 50 países, contando para ello con 140.000 profesionales.

Para la prestación de sus servicios, NTT DATA acompaña a sus clientes para su desarrollo digital con una diversidad de ofertas de servicios con tecnologías de punta en consultoría estratégicas y Advisoring, soluciones informáticas, infraestructura tecnológica, modernización de sus servicios de TI y BPOs. Para la satisfacción y seguridad de los clientes la corporación dispone de una vasta experiencia en todos los sectores de la actividad económica con un gran now how de las geografías en donde presta los servicios.

Con la finalidad de apoyar en el desarrollo de una comunidad de personas única y abierta, respaldada por valores compartidos, que se han ido acrecentando como una inmensa red de talento colectivo orientado a incrementar sus capacidades y conocimientos, para atender con eficiencia a los requerimientos cambiantes de todos nuestros clientes y prevenir con inteligencia el futuro.

Como corporación tenemos la orientación de seguir construyendo un escenario único de colaboración, de manera creativa, cercana y generosa, para seguir evolucionando juntos como una comunidad y sociedad, para la prosperidad día a día en la que existen un contexto cambiante con mejores opciones.

La corporación NTT DATA, tiene presencia en nuestro país a través de cuatro (04) sucursales para la atención de los clientes como se presenta en la Fig. 1.

### Nuestras oficinas

**NTT DATA Dean Valdivia**  
Calle Dean Valdivia 148, P 4  
Edificio Platinum  
San Isidro  
15046  
☎ [+5112038200](tel:+5112038200)

**NTT DATA Blue Tower**  
Calle Portocarrero 262  
La Victoria  
Lima  
15034  
☎ [+5116148000](tel:+5116148000)

**NTT DATA Bloom**  
Avenida Javier Prado Oeste  
2501 second line  
Magdalena del Mar  
15076  
☎ [+5116042130](tel:+5116042130)

**NTT DATA Pentamall**  
Avenida Mansiche 1733  
Trujillo  
Trujillo  
13011  
☎ [+5144604060](tel:+5144604060)

Figura 9. Sucursales de NTT DATA

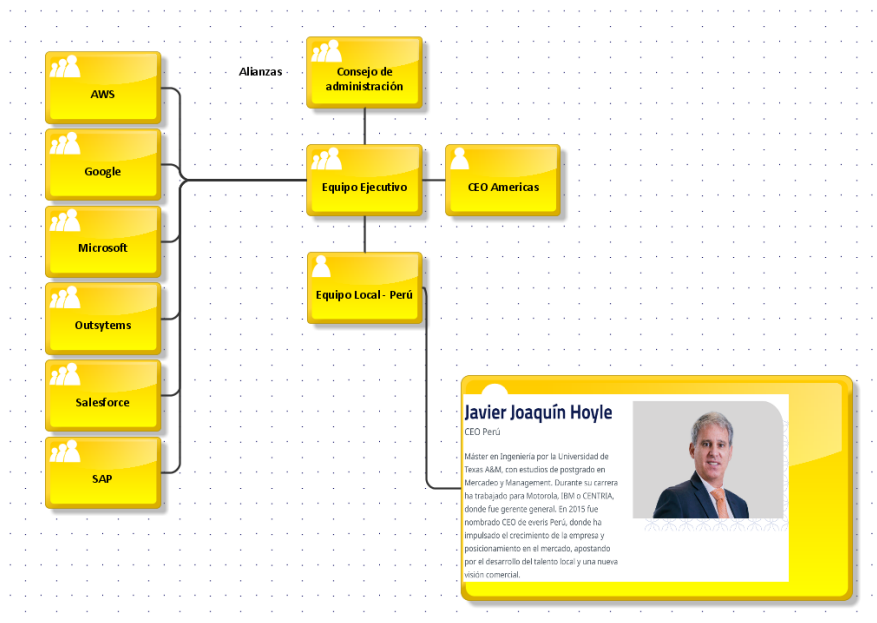


Figura 10. Corporación NTT DATA – Elaboración propia

Fuente NTT DATA <https://pe.nttdata.com/leadership>

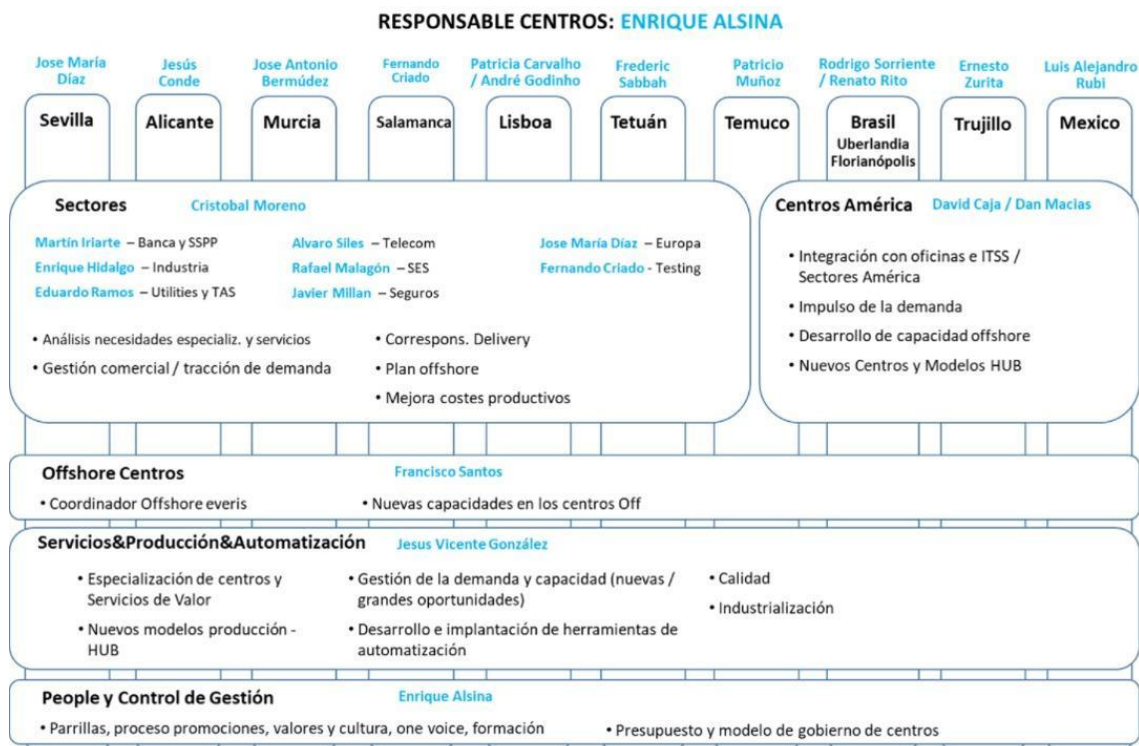


Figura 11. Estructura orgánica de NTT DATA

Se precisa que la empresa **Everis** es parte del Grupo empresarial de **NTT DATA Company**, lo cual se puede corroborar con el contrato del anexo 02. A través de la empresa se realiza el desarrollo de la mejora del aplicativo Tunki de banco Interbank.

## Capítulo II: TRAYECTORIA PROFESIONAL

La experiencia profesional de la señorita Leonela Quispe se ha desempeñado en la empresa diversas labores en el área de tecnologías de información en el campo de la calidad del servicio de la empresa NTTDATA Trusted Global Innovator (<https://pe.nttdata.com/about-us>), para uno de nuestros clientes como INTERBANK, se le ha designado esta labor por la responsabilidad que conlleva atender a clientes de este nivel y por las capacidades del personal de nuestra empresa.

Primer periodo: 15-03-2021 hasta el 30-09-2021 (6 meses 15 días)

Cliente: Interbank – proyecto Plataforma Digital

Rol : QA – Automation Quality assurance – automatización del Aseguramiento de la calidad  
Automatización de las pruebas de para asegurar la calidad de la Plataforma Digital: Me encargaba de automatizar las apis, para probar escenarios de error, así como el famoso Happy Path, esto para poder reducir el tiempo de espera y hacerlo de manera más ágil,

Uso de Herramientas:

Gitbash: para poder realizar lo que es git status, el cual nos ayuda a visualizar los cambios que se están realizando al automatizar

Bitbucket: es el repositorio utilizado donde se tenían las automatizaciones por el lado de QA, como de los demás frentes tales como front y back.

Postman: se utilizaba para las pruebas de apis, aquí se visualizaba los status obtenidos, el body, header entre otros

Gradle: permite la construcción de código para las automatizaciones

Gherkin: lenguaje de especificación del comportamiento esperado, el cual también es utilizado para la comprensión de los tres amigos (negocio, desarrollo y QA)

ScreenPlay: es el patrón de pruebas automatizadas utilizadas, ya que promueve que se tenga buenos hábitos como el de tener organizado y que sea fácil de mantener.

IntelliJ IDE: es el entorno de desarrollo, nos permite poder trabajar con diferentes lenguajes.

Cucumber: este software permite el desarrollo basado en el comportamiento, entre otras.

Segundo Periodo: 01-10-2021 hasta la actualidad

Cliente: Interbank – proyecto Tunki

Rol : QA Tester

Pruebas funcionales: estas pruebas son con la finalidad de verificar que el software funcione como lo esperado, aquí incluye los escenarios de error como happy path.

Automatización mobile: me encargo de automatizar los flujos en la app, para probar escenarios de error, así como el famoso Happy Path, esto para poder reducir el tiempo de espera y hacerlo de manera más ágil,

Pruebas de servicio: estas pruebas se realizan a las apis, utilizando la herramienta de Postman, son con la finalidad de verificar que el servicio responda como lo esperado, de igual manera incluye los escenarios de error como happy path.

Pruebas de integración: estas pruebas las realizamos antes del pase a producción para verificar que la integración de diferentes módulos haya sido transparente y no haya generado algún error.

Pruebas de regresión: en estas pruebas ejecutamos los escenarios ya validados anteriormente y que respondieron de manera correcta.

Pruebas de humo: la realizamos cuando nos generan un nuevo apk, para verificar que la funcionalidad principal esté correcta, esto se hace de manera rápida.

Uso de herramientas:

Jira: para administrar las HUs visualizando las subtareas que realizamos, así como los estados en el que se encuentra.

Testlink: administramos los casos de pruebas, así como sus estados por HU

Appium: nos ayuda para la automatización, de aquí sacamos los selectores

Postman: se utilizaba para las pruebas de apis, aquí se visualizaba los status obtenidos, el body, header entre otros

App tester: aquí se sube las versiones de apk e ipa por el lado de los desarrolladores, a los cuales nosotros podemos acceder para poder validar.

Xray: es la nueva herramienta que se está implementando para la administración de los casos de pruebas de las HUs

Example mapping: es una técnica que nos permite poder tener un mejor entendimiento de la HU, haciendo que veamos soluciones ante un problema utilizando ejemplos antes las reglas(criterios) presentadas, además de la interacción del equipo, por otro lado, permite resolver dudas, ya que tiene una estructura organizada, entre otras.

A pesar de que en su formación profesional la señorita Leonela Quispe no se enseña la asignatura de Aseguramiento de calidad del desarrollo de un software, esta formación se realizó de forma complementaría después de sus estudios universitarios.

## Capítulo III: APLICACION PROFESIONAL

### 3.1. Situación problemática

Aplicativo perteneciente a Interbank, que te permite vincular una Tarjeta Débito Interbank o abrir una billetera 100% digital vinculada a una Cuenta Simple Interbank sin tener que ir al banco. Donde te facilitan realizar diversas operaciones como recargas, pagos de servicio, plineos, pagos a niubiz e izipay, etc. Ya que está enfocada en usuarios nuevos como a micro-comercios, buscando así ser una app amigable y entendible para ellos. El problema se da que el aplicativo no está satisfaciendo las necesidades de los usuarios y que se corroboran con las diversas quejas presentadas por los usuarios expresadas en los canales digitales y que se pueden comprobar en el link siguiente:

[https://play.google.com/store/apps/details?id=pe.com.interbank.mpay.commerce&hl=es\\_419&pli=1](https://play.google.com/store/apps/details?id=pe.com.interbank.mpay.commerce&hl=es_419&pli=1)

A continuación, se presentan muestras de la necesidad



Figura 12. Problemática detectada

La situación problemática del aplicativo requería de realizar las mejoras al aplicativo en tal sentido el proyecto de solución a esta problemática es hacer una reestructuración de este y que se plantea en el punto 3.2.

### 3.2. Proyecto desarrollado

Las pruebas desarrolladas para la aplicación **Tunki** del banco Interbank han sido las siguientes:

1. Pruebas funcionales: estas pruebas las realizamos en la app de “**Tunki**” para verificar que las funcionalidades se comporten según los criterios dados.
2. Pruebas de integración: Lo realizamos constantemente ya que hay Historias de Usuario-HUs donde se integran en una sola o también antes de un pase a producción.
3. Pruebas de Humo: generalmente la realizamos cuando realizan un cambio transparente, para verificar que la funcionalidad se ejecuta correctamente.

#### 3.2.1. Herramientas utilizadas

1. **Scrum:** se utilizó como metodología debido a que con ella se permite dar entregables de valor en cada sprint, para ello se sigue las pautas, respetando sus eventos, programados ya en calendario: Planning, Daily, refinamiento, review y retrospectiva. Y que se presentan en las Figuras 9 a la 14, para el Sprint asignado como Q2.

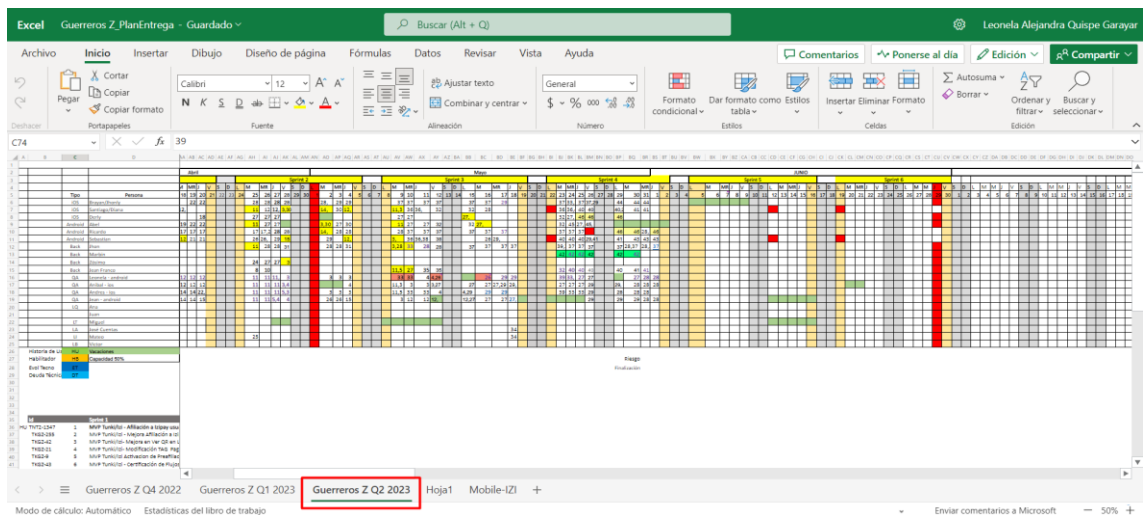


Figura 13. Plan de Entrega del Q2

En la Figura 13, se trabaja el Q2, el cual engloba 6 Sprints, donde se mapea las HUs, responsables, días en la cual se va a trabajar en ellas, feriados vacaciones, etc., todo eso se realiza en los eventos de las Figuras 14 a la 18.

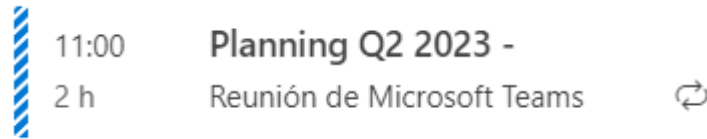


Figura 14. Planeamiento del Q2

En la Figura 14, se presenta la planificación en la reunión por medio de la plataforma Microsoft Teams, para la planificación del Q2. Cuya duración fue de 2 horas. Se asigna a las HU a los responsables quienes la verán, además de que días se estaría viendo ello según la estimación dada.



Figura 15. Daily del Q2

En la Figura 15, se presenta la ejecución del Daily o reunión diaria para determinar que inconvenientes se tuvo, que necesidades se requieren, vale decir el evaluar lo que se hizo en el día anterior, que verá hoy y si tuvo algún impedimento, como se puede comprobar es una reunión de solo 15 minutos.

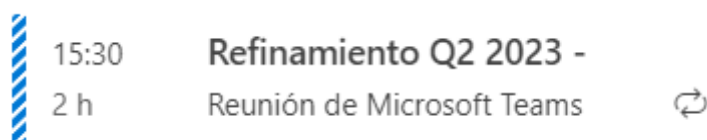


Figura 16. Refinamiento del Q2

En la Figura 16, para el refinamiento del Q2, con un tiempo de trabajo de 2 horas, aquí el PO, presenta y explica las HU, con los criterios de aceptación, figma, y example mapping, los frentes hacen llegar sus consultas respecto a las HUs éstas se van mapeando, respondiendo y añadiendo en el figma como en el example mapping.



Figura 17 de la Retrospectiva Q2

En la Figura 17, en cuanto a la retrospectiva una vez finalizado el Sprint Q2, reunión de 2 horas de duración. Se realiza al finalizar cada sprint, aquí iniciamos con una dinámica para luego ver cómo nos fue en el Sprint, que nos ayudó a avanzar, que nos detuvo, que riesgo se pueden a presentar en un futuro y que nos hizo sentir bien.

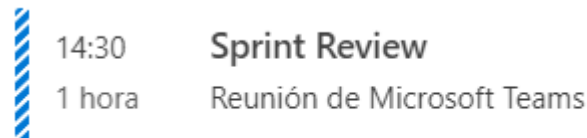


Figura 18. Review del Sprint.

En la Figura 18, como se desprende de la Figura, Se hizo la reunión para la revisión del Sprint desarrollado, reunión que tuvo como duración 1 hora. En este evento presentamos lo certificado, se considera Happy Path, así como escenarios de error que se hizo durante las pruebas.

- Mural:** Herramienta usada en eventos, permite crear un espacio para poder colaborar y contribuir, brindando así ideas para posibles soluciones de manera innovadora.

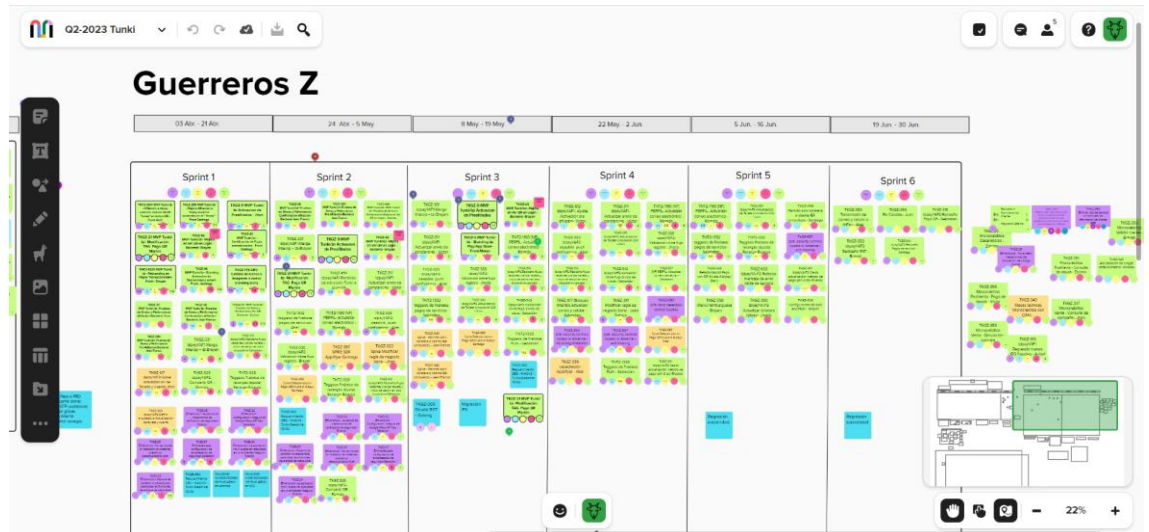


Figura 19. Mural usado para refinamiento y planning de los del Sprint.

En la Figura 19, se muestra lo que se va a trabajar en el Q2, especifica las HU por sprint, el código de estas, las estimaciones, que se toman en consideración en el refinamiento para luego trabajarla en la planning y de la mano del plan de entrega.

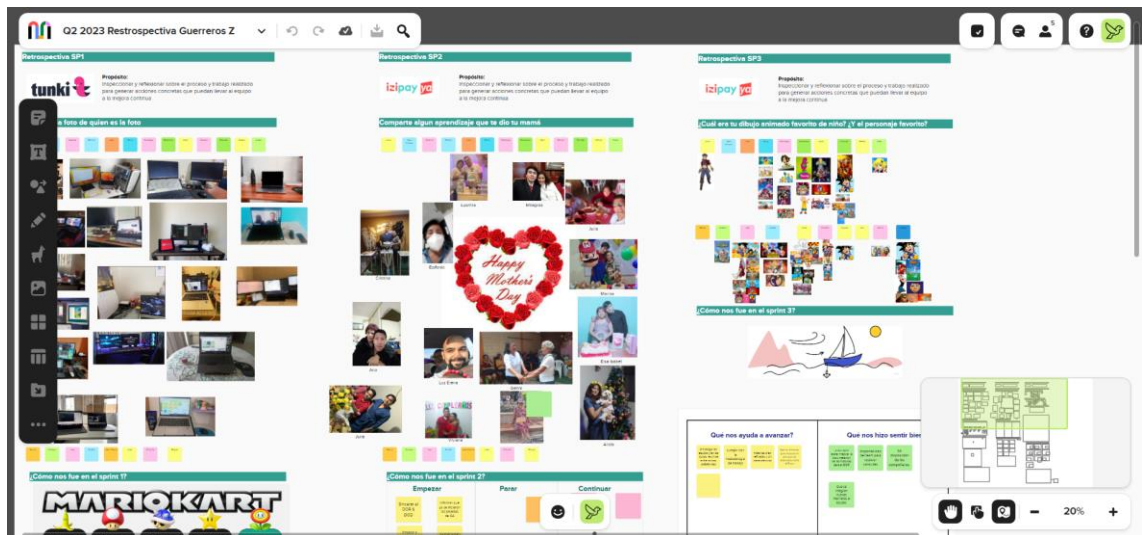


Figura 20. Mural usado para la retrospectiva de cada Sprint.

En la Figura 20, se muestra una pequeña dinámica, para luego entrar a un espacio donde podamos contribuir analizando de lo que fue el sprint, para poder trabajar en lo que hemos aplicado y nos fue bien, que futuros riesgos hemos identificado y cómo podemos actuar sobre ellos.

3. **Jira:** para administrar las HU (Historias de usuarios), usando reglas de negocio para describir los criterios de aceptación, con ello viene de la mano de estados para saber cómo va el avance, añadimos subtareas, además de crear tickets para reportar issues, poner tiempo que toma ejecutar, entre otros



Figura 21. Especificación de la HU de un usuario del Tunki

En la Figura 21, se puede apreciar la importancia del Jira, para el control y seguimiento de proyecto, en la que se presenta una historia de usuario usando regla de negocio, con sus criterios de aceptación, para el desarrollo e implementación.

	TUNKI-4996	[iOS] - TUNKI-4996: Agregar pantalla de confirmación al...		FINALIZADO
	TUNKI-5012	[Android] Actualización de prepare		FINALIZADO
	TUNKI-5015	[Android] Implementación de valor min y max desde co...		FINALIZADO
	TUNKI-5023	[Android] Actualización de otp		FINALIZADO
	TUNKI-5041	[Back] refactor generate OTP		FINALIZADO
	TUNKI-5021	[Back] add limit config to db		FINALIZADO
	TUNKI-5013	[Back] refactor on confirmOTP		FINALIZADO
	TUNKI-5042	[Back] refactor prepare OTP		FINALIZADO
	TUNKI-5044	[Back] solve comments about pullRequest		FINALIZADO
	TUNKI-5052	[Back] despliegue en UAT de MS: tunki-bill-payment		FINALIZADO
	TUNKI-5024	[iOS] Agregar el desplegable a la pantalla de recargas		FINALIZADO

Figura 22. Gestión de las tareas finalizadas en Jira

En la Figura 22, se aprecia las subtareas creadas por cada frente para especificar que se está trabajando, aquí ponemos el responsable, vamos cambiando los estados en el que se encuentran hasta llegar a “finalizado”, esto para monitorear y controlar el avance que se viene realizando, la herramienta Jira ayuda al control y seguimiento.

	TNT2-905	[iOS] Error en redacción en la pantalla Cerrar mi Billet...		FINALIZADO
	TNT2-906	[Back] Error de redacción Pop-Up Tunki Tu Billetera T...		FINALIZADO
	TNT2-933	[Back] Al cerrar mi cuenta billetera muestra un popup ...		FINALIZADO
	TNT2-934	[Android] Errores		FINALIZADO
	TNT2-1052	[Android] constancia no muestra el color correcto		FINALIZADO
	TNT2-1054	[Android] se muestra de diversos tamaños		FINALIZADO
	TNT2-1261	[Android] se muestran los bordes más opacos en la c...		FINALIZADA

Figura 23. Issues encontrados durante las pruebas dentro de una HU de Tunki

En la Figura 23, representa los issues que se van encontrando mientras se ejecutan los casos de pruebas contemplados, para ello se detalla en que SO (sistema operativo) se encontró o si fue de Back, se añade un título, dentro una descripción y datos necesarios

para replicar el escenario, se le asigna al frente responsable y de igual manera tenemos estados para monitorear el avance.

- 4. Figma:** es una herramienta de diseño colaborativo que permite a los equipos crear prototipos de interfaces de usuario de manera eficiente. Permite la creación y edición en tiempo real, facilitando la colaboración y la retroalimentación entre los miembros del equipo.

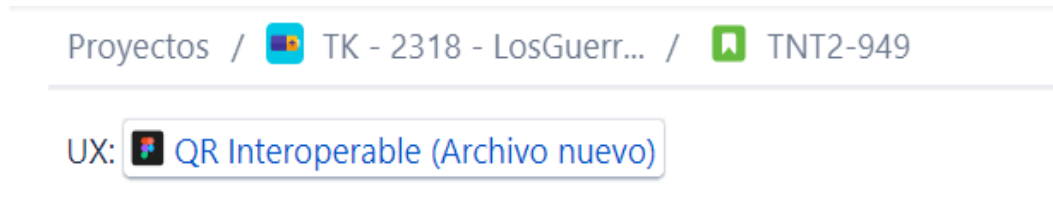


Figura 24. Historia de usuario para QR interoperable usando figma

En la Figura 24, representa el link del flujo que veremos, aquí es donde encontraremos lo que describe la HU en sus criterios de aceptación, y como vemos haciendo referencia a la nueva herramienta utilizada.

Anteriormente se utilizaba **invision** para ver los diferentes flujos de las pantallas que se van a rediseñar o a implementar, esto también añadidas al proyecto y la HU correspondiente. A la fecha ya se ha estandarizado el empleo de la herramienta de Figma para el diseño.

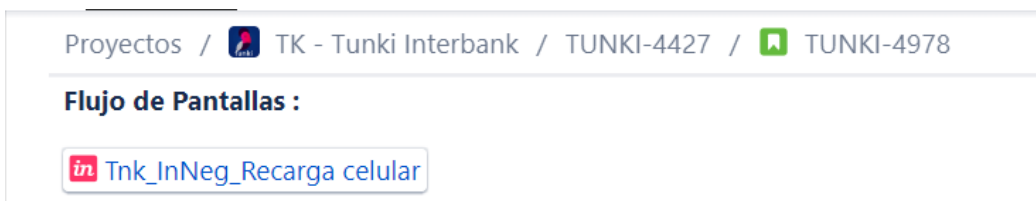


Figura 25. Historia de usuario para la recarga del celular usando Invision

En la Figura 25, representa el link del flujo de recarga de celular, la cual veremos, aquí es donde encontraremos lo que se describe la HU en sus criterios de aceptación, y como vemos haciendo referencia a la herramienta que antes se usaba.



## \* Figma (actual)

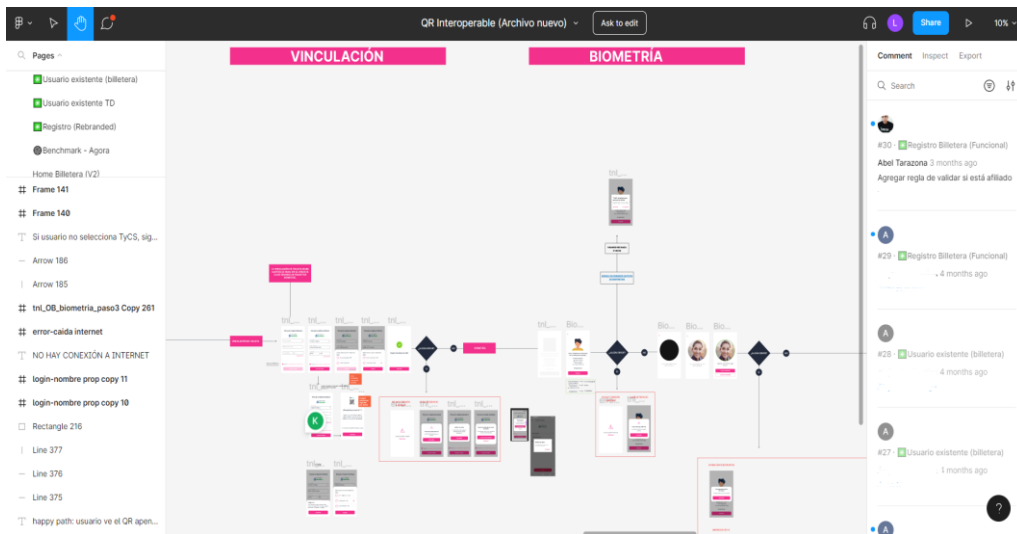


Figura 28. Rediseño sin Rebranding con la herramienta figma

En la Figura 28, nos muestra rediseño de la HU de vinculación de tarjeta, esto se realizó antes del rebranding (cambio de la marca Tunki-izipayYA)

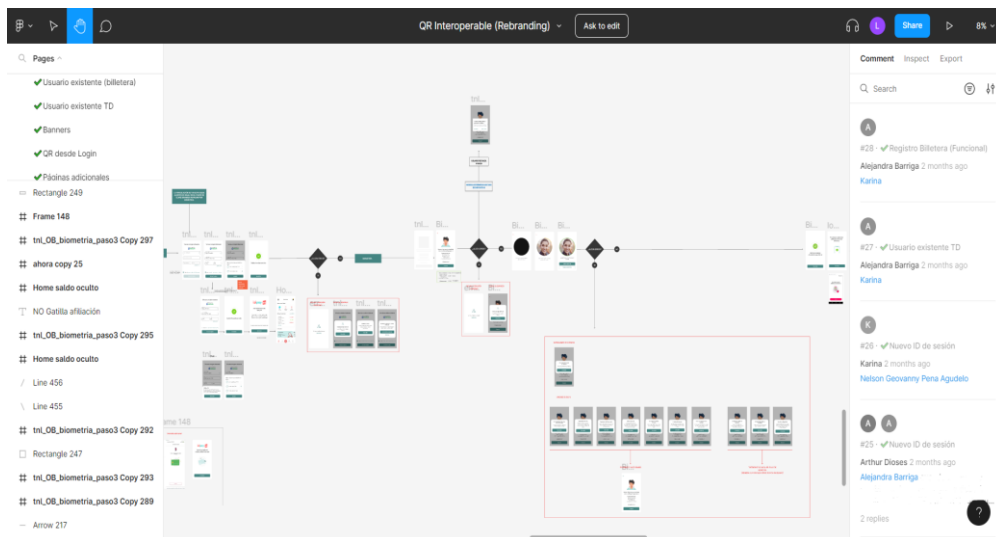


Figura 29. Rediseño con Rebranding con la herramienta figma

En la Figura 29, nos muestra rediseño de la HU de vinculación de tarjeta, esto se realizó ya con el rebranding (cambio de la marca Tunki-izipayYA), usando la herramienta actual que es figma

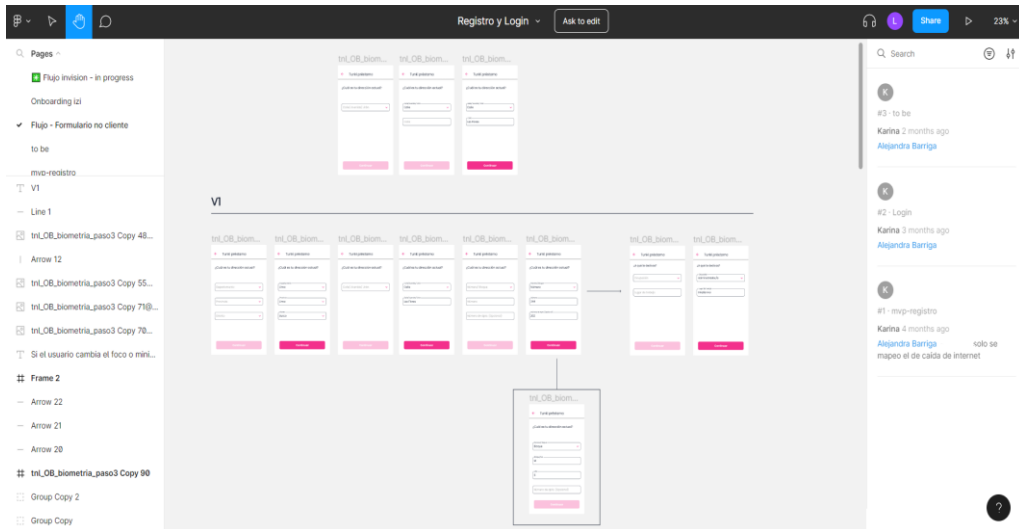


Figura 30. Desarrollo sin Rebranding con la herramienta figma

En la Figura 30, representa el desarrollo de la HU formulario no cliente, esto se realizó sin el rebranding (cambio de marca Tunki-izipayYA), usando la herramienta actual que es figma

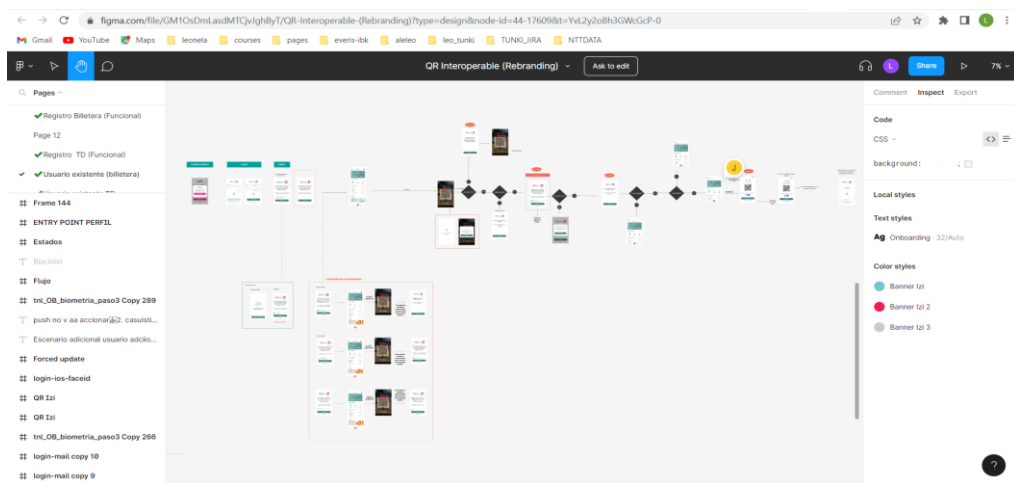


Figura 31. Desarrollo con Rebranding con la herramienta figma

En la Figura 31, representa el desarrollo de la HU QR interoperable, esto se realizó ya con el rebranding (cambio de marca Tunki-izipayYA), usando la herramienta actual que es figma

**5. Postman:** Postman es una plataforma de desarrollo de API que permite a los desarrolladores probar, documentar y compartir APIs de manera sencilla. Proporciona

un entorno amigable para enviar y recibir solicitudes HTTP, lo que facilita el desarrollo y la depuración de API.

Para testear endpoints, usando diferentes métodos como GET, POST Y PUT, para obtener los diferentes códigos de respuesta según lo esperado, usando las variables de entorno y global, etc .

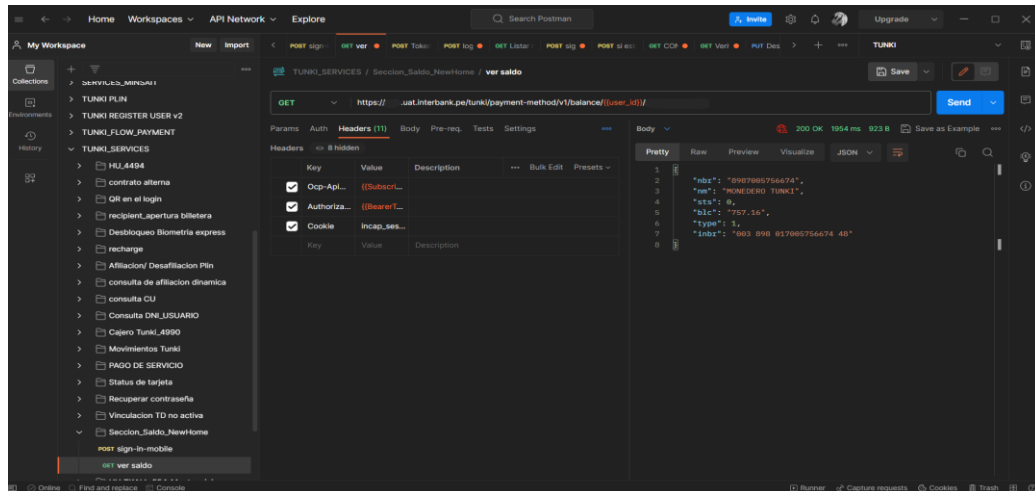


Figura 32. Testeo con Postman con el método Get

En la Figura 32, vemos una prueba para la HU de sección ver saldo para un usuario billetera, el método, endpoint, header, response y respuesta que obtenemos al ejecutarlo.

- 6. IDE IntelliJ:** es un entorno de desarrollo integrado utilizado principalmente para desarrollar aplicaciones en lenguajes de programación como Java, Kotlin y Scala. Proporciona características avanzadas de edición de código, depuración y gestión de proyectos, lo que aumenta la productividad de los desarrolladores. Es el software que se utiliza para desarrollar y realizar las automatizaciones de los flujos existentes de las aplicaciones

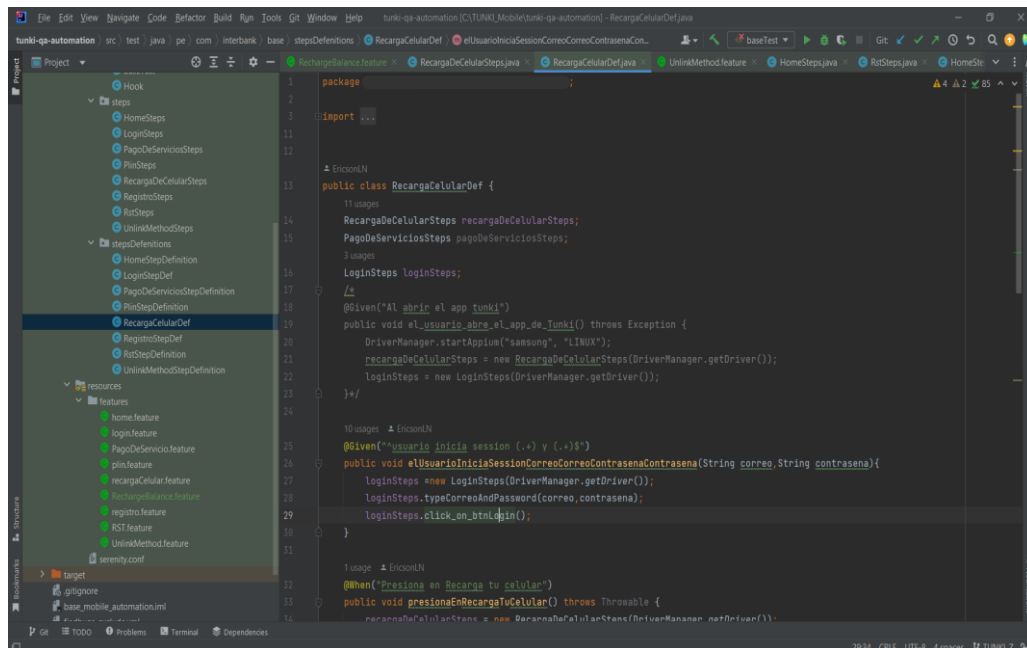


Figura 33. Automatizador de los flujos en las aplicaciones

En la Figura 33, representa la herramienta utilizada para llevar las automatizaciones de algunos flujos de la app Tunki, aquí la vemos el patrón de diseño aplicado, variables utilizadas para utilizarse en las clases.

**7. BEHAVIOR DRIVEN DEVELOPMENT (BDD):** es una metodología de desarrollo de software que se enfoca en la colaboración entre los diferentes roles en un equipo de desarrollo. Utiliza el lenguaje Gherkin para describir el comportamiento esperado de una aplicación y Cucumber como una herramienta para ejecutar esos escenarios de comportamiento.

En español desarrollo guiado por el comportamiento, nos permite un trabajo en conjunto, para ello plantea un lenguaje en común, tanto para la parte técnica como para la parte del negocio.

El funcionamiento sería de la siguiente manera:

- ✓ Analizar los requisitos, consideraciones, objetivos y desempeño que debe cumplir
- ✓ Describir los escenarios que se van a presentar, para anticiparse ante la respuesta que se quiere obtener.
- ✓ Por último, se obtiene la respuesta esperada, según lo realizado anteriormente.

**8. Example Mapping:** Nos ayuda a entender la Historia de Usuario, además de dar a conocer ciertos requisitos que no se están considerando y así ver que se nos puede estar escapando de la mano, para ello tenemos las siguientes partes:

- ✓ El nombre de la HU: es lo que va primero y normalmente va en un recuadro de amarillo
- ✓ Reglas: viene a ser los criterios de aceptación que se debe cumplir, normalmente se ponen en recuadros celestes
- ✓ Ejemplos: vienen a ser los escenarios que se pueden presentar en por cada regla descrita, esto ayuda tanto a los 3 amigos (negocio, desarrollo y testing), para ello usamos gherkin, en el caso de los escenarios, ya que debe llevar una precondición, luego la acción y el resultado esperado
- ✓ Preguntas: son las que surgen luego de explicar

En algunos casos cuando los ejemplos son más de 5, se divide la regla haciéndolo más específico.

Además, si se extiende con demasiadas reglas, se practica el “SLICING”, que es prácticamente dividir la HU (Historia de Usuario) para poder dar un entregable durante el sprint.

Por último, cuando hay muchas card rosadas, quiere decir que aún no se tiene todo el conocimiento de la HU, por lo que se lleva la HU para resolverlas.

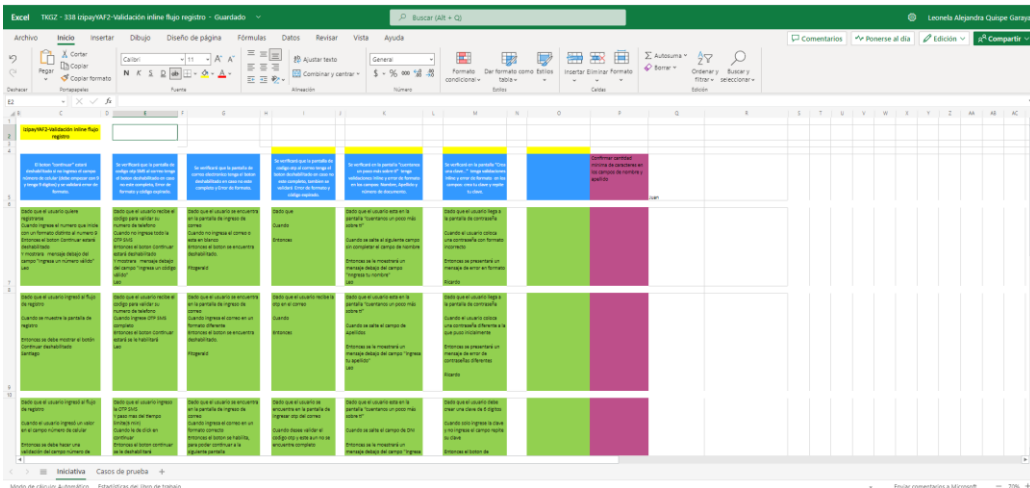


Figura 34. Mapeo de la HU y verificar requisitos

En la Figura 34, nos muestra la ejecución correcta de un ejemplo, que se aplicó para una HU, siguiendo la estructura y consideraciones impuesta para ello, como son los colores de las tarjetas especificadas líneas más arriba.

**9. Gherkin:** Es el lenguaje de negocio que se utiliza para que sea entendible para los tres amigos: negocio, desarrollo y testing.

Algunos elementos que se usan son:

- ✓ Feature: Nos indica el nombre de la funcionalidad que vamos a probar
- ✓ Scenario: Nos va a describir el escenario que se va a probar
- ✓ Given: es la pre condición de que debe cumplir
- ✓ When: la acción que se va a realizar
- ✓ And:
- ✓ Then: Muestra el resultado esperado

```
@test_RecargaDeCelular
> Feature: Iniciar sesion en Tunki para realizar una recarga de celular
  Como usuario de Tunki
  Quiero iniciar sesion
  Para realizar una recarga de celular.

Background:
  Given Al abrir el app tunki

@test_RecargaConTDoBilletera
> Scenario Outline: Realizar una recarga de celular con Billetera en Tunki
  Given usuario inicia session <Correo> y <Contrasena>
  When Presiona en Recarga tu celular
  And Selecciona el operador <Operador>
  And Escribe el numero de celular <Celular>
  And Escribe el monto <Monto>
  And Oprime en el boton Aceptar
  And Presiona en confirmar
  And Muestra la pantalla de Recarga realizada
  And Presiona en el boton ir al inicio
  Then Se muestra la pantalla del Home
  Examples:
  | Correo | Contrasena | Operador | Celular | Monto |
  | tarjetas.11@gmail.com | 111111 | Claro | 989175757 | 5 |
  ##| lunes7@gmail.com | 111111 | Claro | 989175757 | 5 |
  #Ingresar usuario que tenga método de pago Billetera o TD
```

Figura 35. Flujo redactado en el lenguaje Gherkin

En la Figura 35, representa la redacción de los pasos a realizarse para el flujo de recarga, pero usando el lenguaje Gherkin, cumpliendo la estructura establecida como el de tener los elementos como Given (precondición), When (la acción), Then (resultado esperado), example (para poder reutilizar lo descrito), detrás de ello se ejecutan líneas de código, que harán que se ejecute lo descrito.

**10. Cucumber:** sirve para poder ejecutar las descripciones funcionales que han sido descritas en la Figura 35.

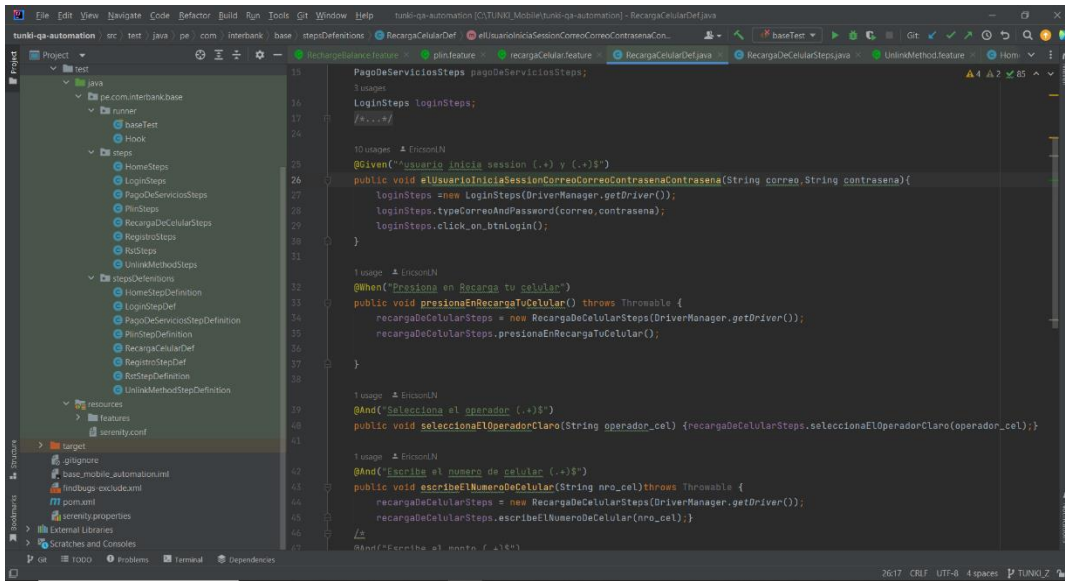


Figura 36. Como ejecuta las descripciones de la Figura 35

En la Figura 36, Representa como ejecuta lo descrito en lenguaje gherkin, podemos apreciar que menciona los elementos Given, And, When y Then descritos en la Figura 31,

**11. Page Object Model (POM):** es un patrón de diseño utilizado en el desarrollo de pruebas automatizadas. Se utiliza para separar la lógica de las pruebas de la estructura de la página, lo que facilita el mantenimiento y la reutilización de las pruebas.

Es el patrón de diseño, que se utiliza para dar mantenimiento y evitar duplicidad de código, ya que nos permite reutilizar código ahorrando tiempo, puesto que si cambia algo solo modificamos los orígenes.

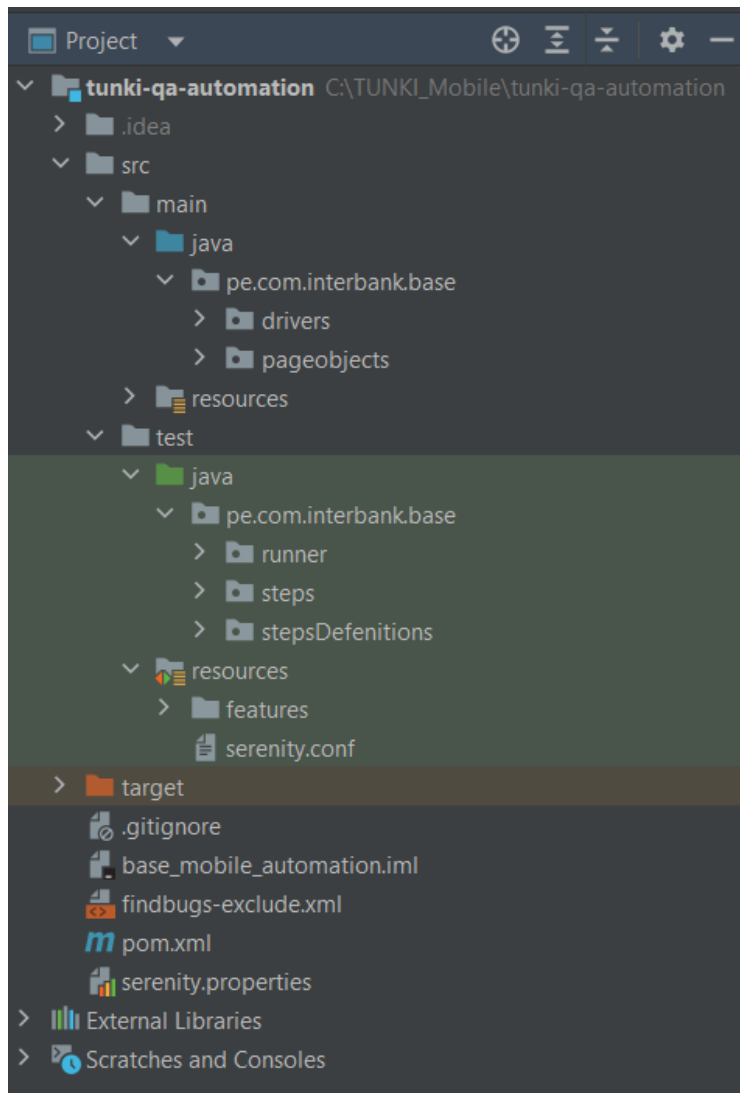


Figura 37. Herramienta para mantenimiento del código

En la Figura 37, Representa la estructura que se sigue para poder dar mantenimiento a nuestro código.

**12.Stetho:** es una herramienta de depuración para aplicaciones Android. Permite inspeccionar y depurar aplicaciones en tiempo real, visualizar la base de datos local y la jerarquía de vistas, y realizar otras tareas de depuración para mejorar la calidad y el rendimiento de las aplicaciones.

Mediante el navegador “**BRAVE**” nos ayuda a debuggear el aplicativo, ¿para aplicativos de **Android**, podemos ver qué servicio consumimos en cada acción que realicemos, además de las respuestas de estas, ayudándonos a ver cuándo se nos presenta un error en la app.

Tenemos las siguientes partes:

- ✓ **Network:** nos dará acceso a los **headers**, **payload** y **preview** de las peticiones al servidor.

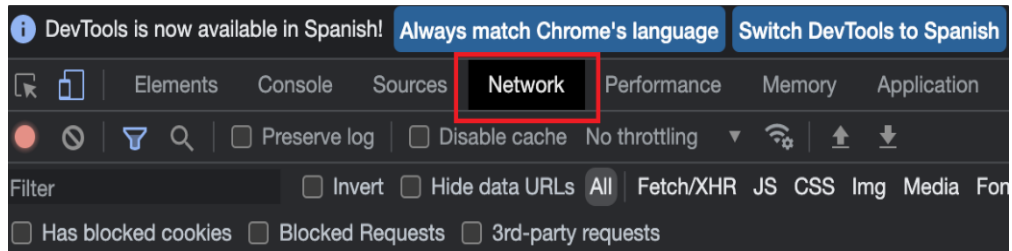


Figura 38. Interfaz de comunicación con el servidor

En la Figura 38, nos da el acceso a los headers, payload y preview al ejecutar las pruebas del flujo que estemos validando.

- ✓ **Name:** Enlista los servicios que va consumiendo en cada acción que realicemos
- ✓ **Header:** Cabeceras que la app envía al servidor en la llamada
- ✓ **Payload:** Esto es lo que la app está enviando al servidor (si en la petición de la app al servidor hay un body)
- ✓ **Preview:** es la respuesta del servidor

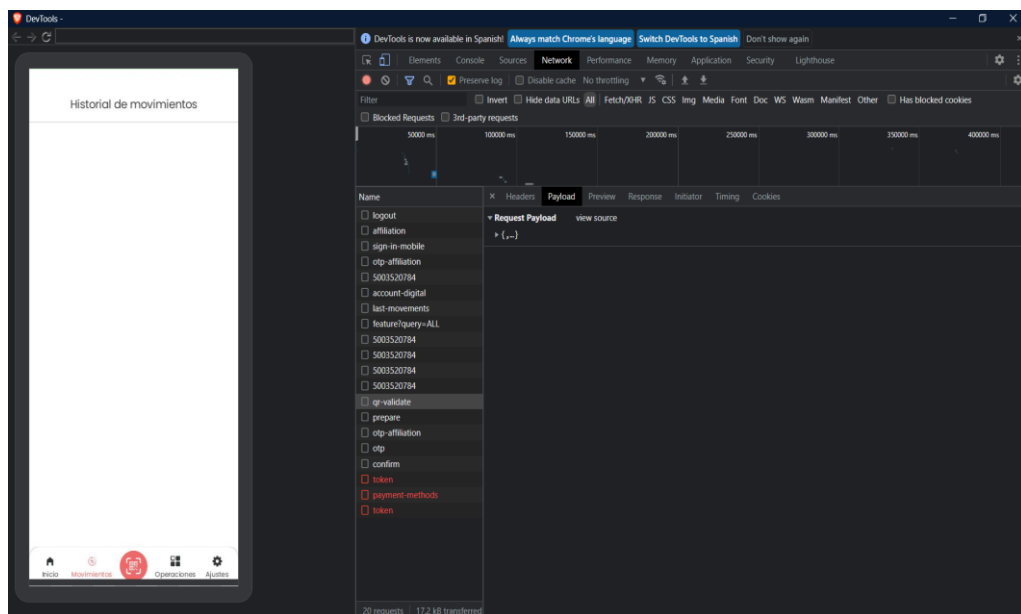


Figura 39. Interfaz de control y monitoreo de los errores que se puedan presentar

En la Figura 39, Nos muestra mientras ejecutamos los casos de pruebas, los headers, payload y preview al realizar cada paso dentro de la App, de esta manera nos permite tener alcance a la información que enviamos y recibimos.

**13. New Relic:** es una plataforma de monitoreo de aplicaciones que ayuda a los equipos de desarrollo a comprender y mejorar el rendimiento de sus aplicaciones. Proporciona información detallada sobre el rendimiento, la disponibilidad y la experiencia del usuario, lo que permite identificar y solucionar problemas rápidamente.

Nos permite monitorizar los tiempos de respuestas, mensajes de error, crash que ocurren al utilizar los servicios en la aplicación. No solo ello, sino que permite hacer análisis estadísticos.

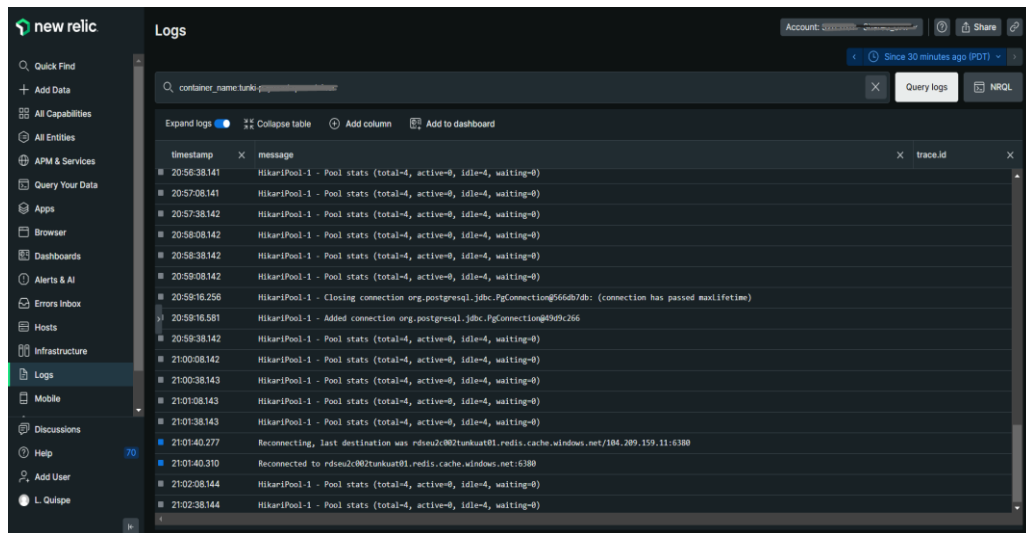


Figura 40. Monitoreo de los tiempos de respuestas y otros parámetros

En la Figura 40, Representa el monitoreo más a detalles de las pruebas realizadas, para obtener información de los errores, request, response, servicio que se consume y/o genera el error, así como información más específica de crash que pueden ocurrir en la app, entre otros.

**14. SoapUI:** es una herramienta de prueba de servicios web que permite a los desarrolladores y probadores realizar pruebas funcionales, de carga y de seguridad en servicios web. Proporciona una interfaz gráfica intuitiva para crear, ejecutar y analizar pruebas de servicios web.

Se usa para poder obtener información de las **TD** o **TC** asignada a usuarios para pruebas, de igual manera para afiliar a banca sms usuarios para las pruebas de las HUs.

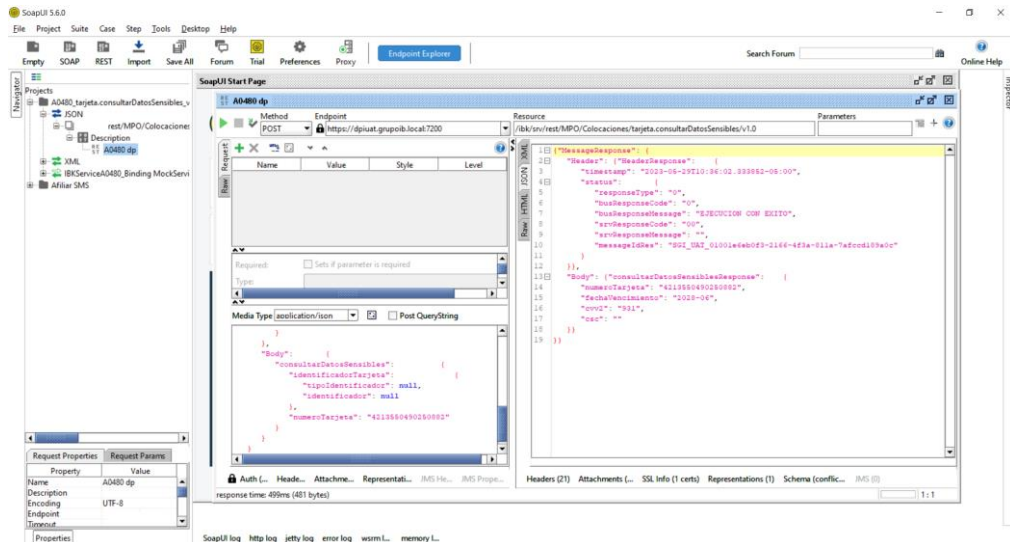


Figura 41. Exploración de los puntos finales para las pruebas de la HUs

En la Figura 41, Nos muestra la ejecución de consulta para obtener los datos de un TD asignada a un usuario creado, usando la herramienta mencionada en la Figura 38, para este caso se debe estar conectado a la VPN, mostrado en la Figura 39.

- ZOC8:** es un emulador de terminal ampliamente utilizado que proporciona conectividad a una amplia gama de sistemas operativos y protocolos. Es utilizado por desarrolladores y administradores de sistemas para acceder y administrar servidores y dispositivos de forma remota.

Una vez activo la VPN, podemos mediante terminales, crear usuarios, asignar TD, crearles cuentas, poner en diversos estados sus cuentas, agregarle saldo, quitarle saldo, modificarlo, etc, todo esto para las pruebas según se requieran.

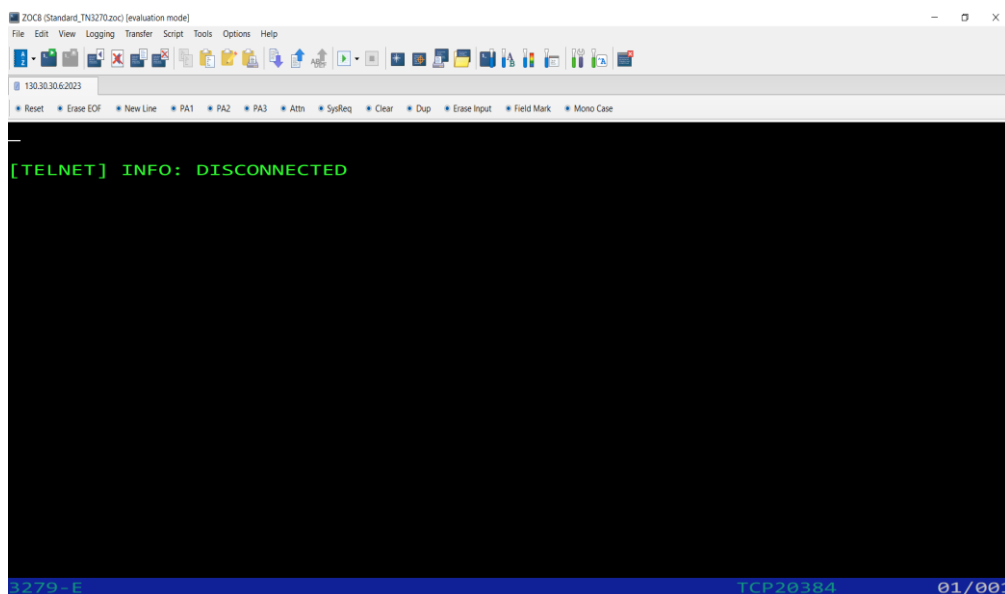


Figura 42. Comunicación mediante la VPN para la realización de las pruebas

En la Figura 42, Representa la herramienta, que conectada a la VPN, en la Figura 39, permite crear usuarios, asignarles cuentas, TD, modificar sus datos, entre otros

- 16. CITRIX:** es una plataforma de virtualización y acceso remoto que permite a los usuarios acceder a aplicaciones y escritorios de forma remota desde cualquier ubicación y dispositivo. Proporciona un entorno seguro y escalable para la entrega de aplicaciones y escritorios virtuales.

Nos permite conectarnos a la VPN, para acceder a ciertas aplicaciones para las pruebas.

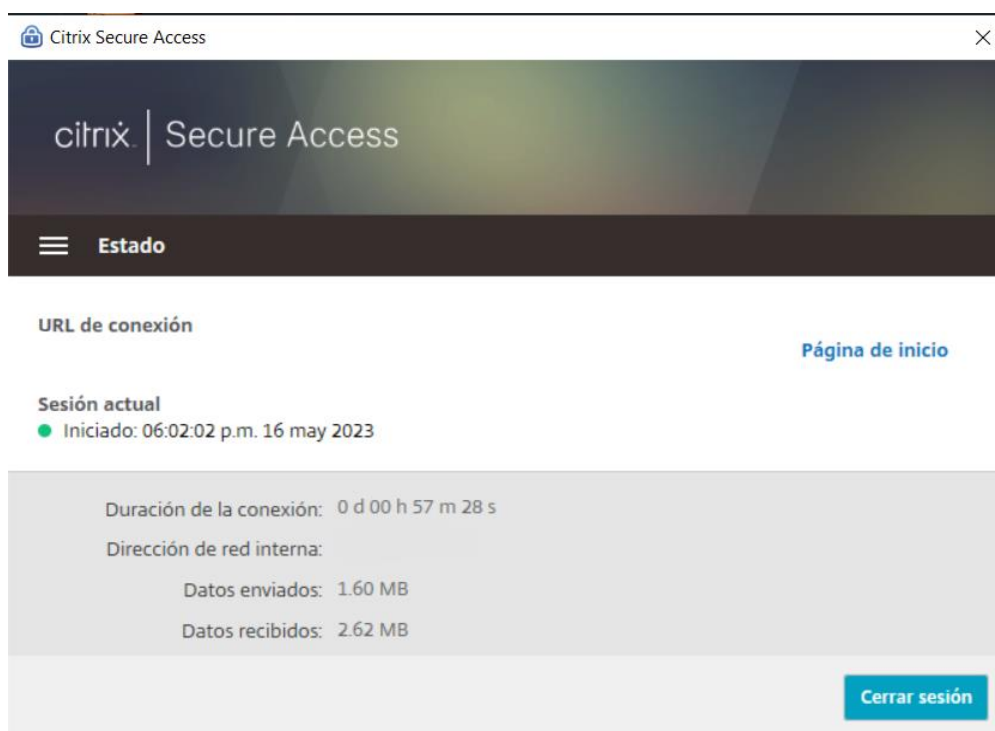


Figura 43. Seguridad de la conexión

En la Figura 43, Representa la conexión a la VPN del cliente para seguridad y poder ingresar a aplicativos utilizados para la ejecución de las pruebas para la app Tunki.

- 17. Appium:** es una herramienta de automatización de pruebas diseñada específicamente para probar aplicaciones móviles en diversas plataformas, como iOS, Android y Windows. Es una opción popular entre los equipos de desarrollo y pruebas debido a su capacidad para escribir y ejecutar pruebas en varios lenguajes de programación, como Java, C#, Ruby, Python, entre otros. Appium utiliza el concepto de "pruebas basadas en el framework de pruebas" (test framework agnostic) para permitir la reutilización de código y scripts de prueba en diferentes plataformas y entornos. Esto significa que los desarrolladores y probadores pueden utilizar

el mismo conjunto de pruebas para probar una aplicación en diferentes dispositivos y sistemas operativos móviles, lo que ahorra tiempo y esfuerzo.

Además, Appium proporciona una interfaz fácil de usar para interactuar con las aplicaciones móviles durante las pruebas. Permite simular acciones como toques, desplazamientos, gestos multitáctiles, entrada de texto, etc. Esto facilita la creación de pruebas automatizadas que imitan el comportamiento de los usuarios reales.

Lo utilizamos para realizar automatizaciones aplicaciones mobile tanto para iOS y Android.

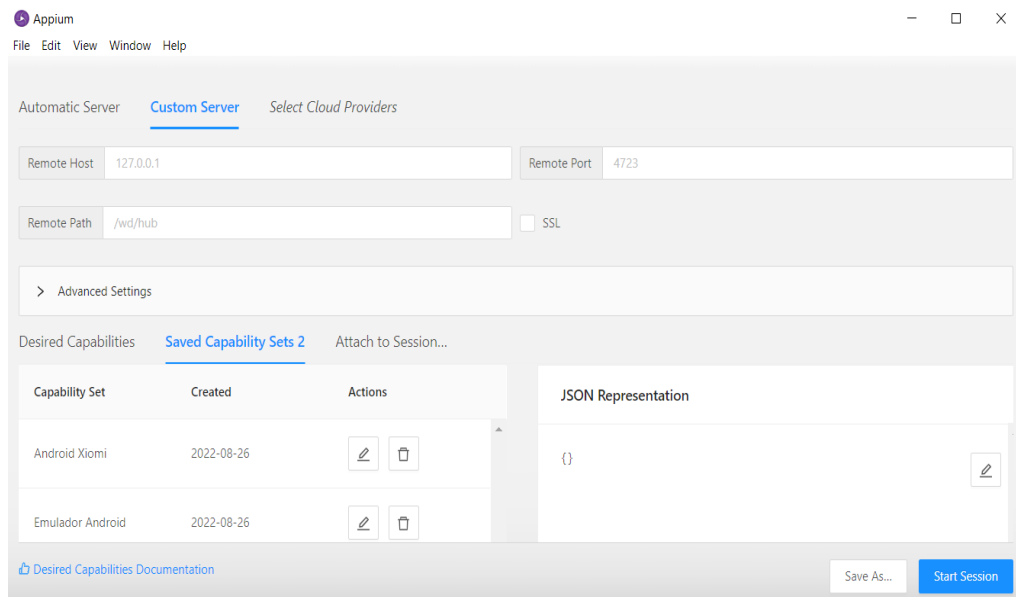


Figura 44. Personalización del servidor para Android

En la Figura 44, nos permite configurar mobile, para poder conectarnos a través de este, de esa manera sacar datos como los localizadores y realizar las automatizaciones que va de la mano de la herramienta de la Figura 34.

**18.PostgreSQL:** es su sistema de gestión de dato, de igual manera lo utilizamos para las pruebas para realizar consultas, modificar status, o ciertos campos como DNI y teléfono, sacar data, etc.

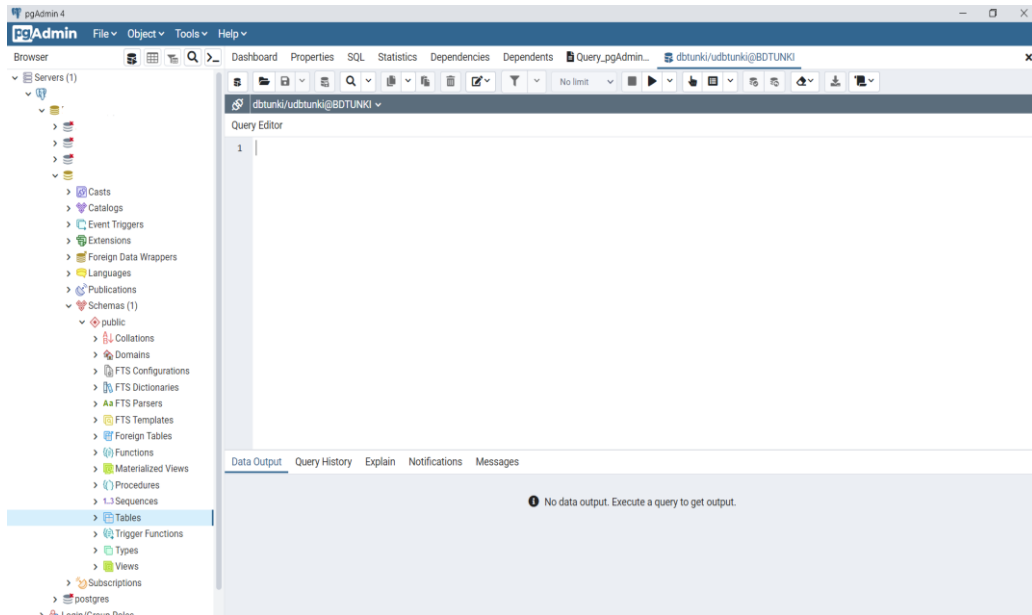


Figura 45. Utilización del gestor de datos Postgres

En la Figura 45, Nos permite acceso a las tablas para realizar consultas y/o modificaciones necesarias según nuestras pruebas.

**19. SAT:** Se utiliza para poder dar de alta y acuse una TC a los usuarios, para las pruebas.

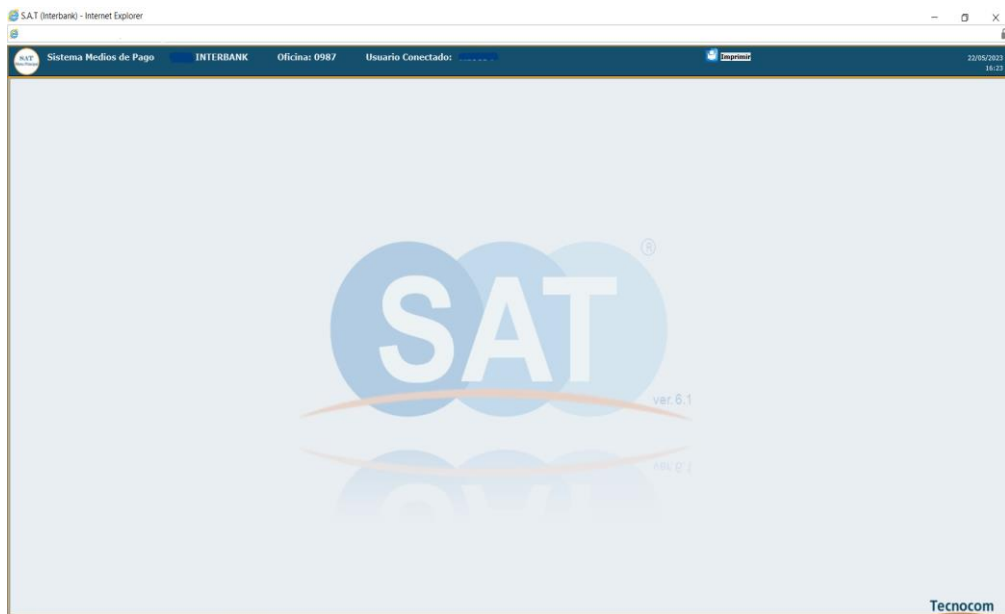


Figura 46. Creación y Acuse de tarjeta de crédito

En la Figura 46, la herramienta SAT, nos permite poder asignarle y acuse de la tarjeta de crédito para usuarios, creados para nuestras pruebas.

20. **SPU:** Nos permite consultar a los usuarios, mediante su DNI, CE, pasaporte, número de tarjeta o código único, de esta manera obtendremos información sobre el método de pago del usuario, además de la información de cada uno de esos métodos.

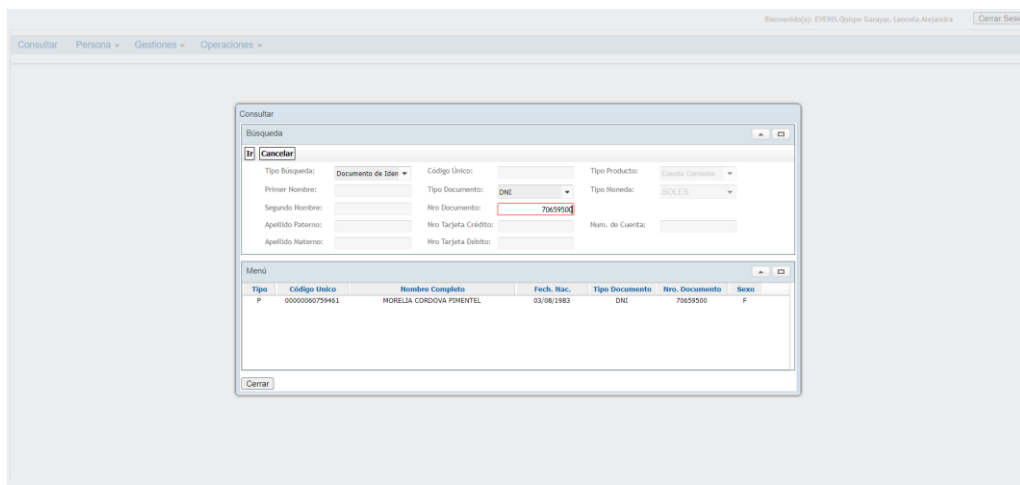


Figura 47. Consultas mediante SPU

En la Figura 47, Nos permite consultar al usuario, y obtener la “pan” de su TD o TC, según con lo que cuente el usuario, además el estado de este.

21. **Testlink:** es una herramienta de gestión de pruebas de código abierto que se utiliza para planificar, diseñar, ejecutar y realizar un seguimiento de las pruebas en proyectos de desarrollo de software. Proporciona una interfaz web intuitiva que permite a los equipos de desarrollo y pruebas colaborar y administrar eficientemente sus actividades de pruebas.

TestLink permite la creación de casos de prueba detallados, que incluyen pasos, precondiciones y postcondiciones. Además, permite agrupar casos de prueba en suites y asignarlos a diferentes versiones o ciclos de prueba. Esto facilita la organización y la ejecución sistemática de las pruebas.

La herramienta también proporciona funciones para rastrear y gestionar los resultados de las pruebas, generar informes y gráficos de cobertura, y realizar un seguimiento de los problemas o defectos descubiertos durante las pruebas. Esto permite una gestión eficiente del ciclo de vida de las pruebas y una mejor visibilidad del progreso y la calidad del software en desarrollo.

Aquí administramos las evidencias de las pruebas ejecutadas, aquí encontraremos estatus, como pasado, fallado o desestimado, para este último agregamos un comentario del por qué se desestimó, además añadimos el Word donde contiene las evidencias, nos permite ver por sprint y quien fue quien lo certificó.

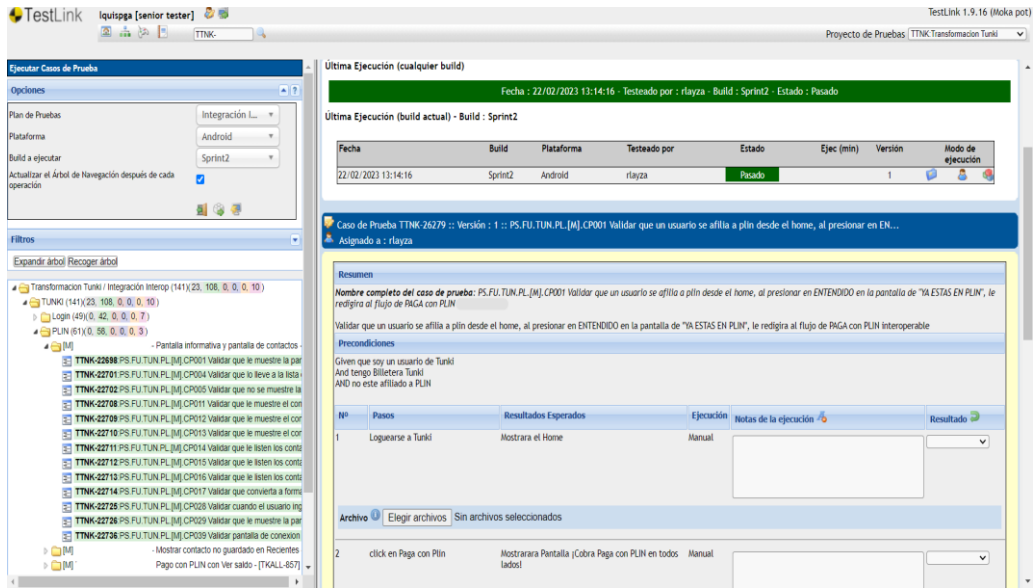


Figura 48. Monitoreo y ejecución mediante Testlink

En la Figura 48, Nos permite monitorear el estado de los casos de pruebas considerados para el flujo que se está certificando, si se ejecutó de manera exitosa, además se agrega las evidencias de lo que ejecutado cumple con el requisito mencionado, si en caso se desestimara, se añade el comentario del motivo por el cual se ejecutó de esa manera en testlink, de igual manera si fallara.

### 3.2.2 Resultados

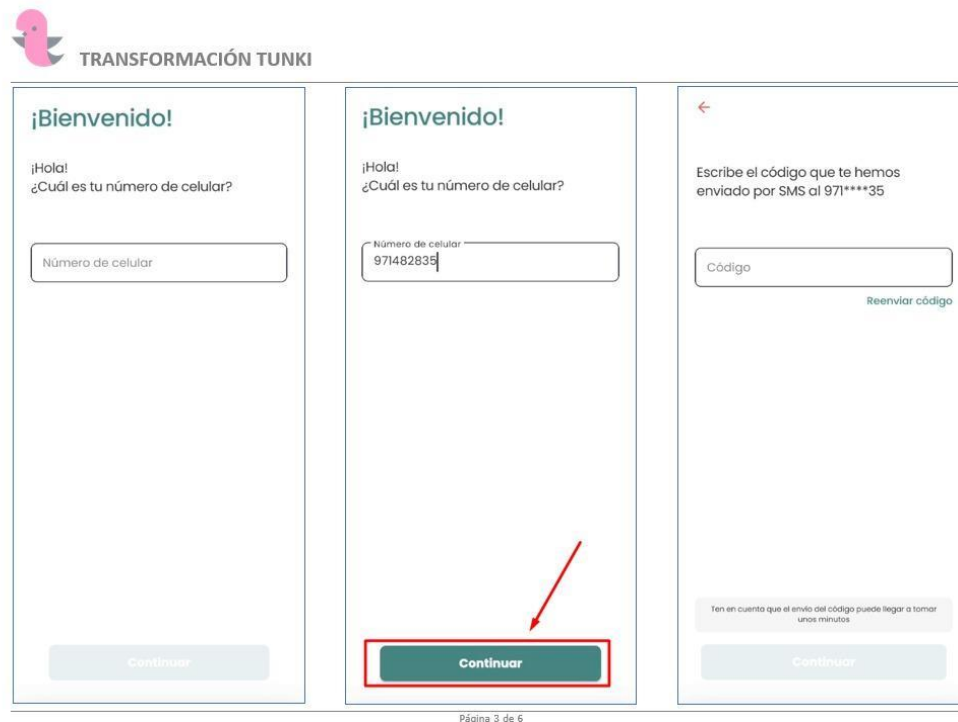


Figura 49. Interfaces de bienvenida

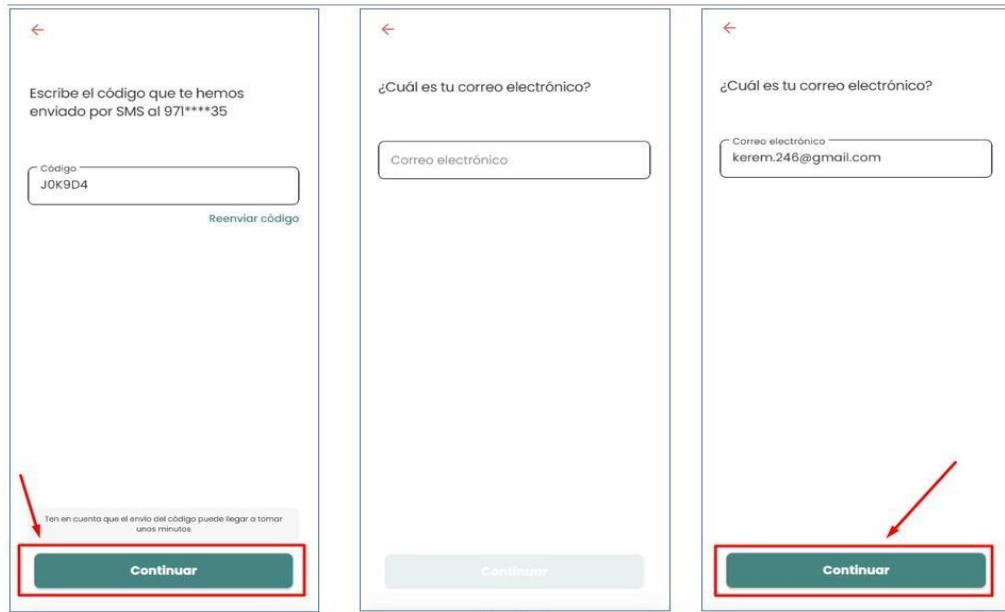


Figura 50. Interfaces seguridad de acceso

DATA EMEAL > tunki > EVIDENCIAS\_TUNKI > Q2\_2023 > Sprint\_4 > TKGZ-338-izipayYAF2-Validación inline flujo registro > DOC

Nombre	Estado	Fecha de modificación	Tipo	Tamaño
TUNKI_TTNK-18600_CP001_Ingresar a Re...	✓	25/05/2023 11:00	Documento de Mi...	211 KB
TUNKI_TTNK-18600_CP002_Ingresar a Re...	✓	25/05/2023 10:38	Documento de Mi...	736 KB
TUNKI_TTNK-18600_CP003_Ingresar prim...	✓	25/05/2023 11:01	Documento de Mi...	180 KB
TUNKI_TTNK-18600_CP004_Ingresar núm...	✓	25/05/2023 11:03	Documento de Mi...	173 KB
TUNKI_TTNK-18600_CP005_Ingresar núm...	✓	25/05/2023 11:03	Documento de Mi...	209 KB
TUNKI_TTNK-18600_CP006_Ingresar núm...	✓	25/05/2023 11:04	Documento de Mi...	172 KB
TUNKI_TTNK-18600_CP007_Ingresar a la ...	✓	25/05/2023 12:11	Documento de Mi...	188 KB
TUNKI_TTNK-18600_CP008_Ingresar la OT...	✓	25/05/2023 11:41	Documento de Mi...	231 KB
TUNKI_TTNK-18600_CP009_Ingresar la OT...	✓	26/05/2023 15:20	Documento de Mi...	423 KB
TUNKI_TTNK-18600_CP010_Ingresar la OT...	✓	26/05/2023 15:20	Documento de Mi...	274 KB
TUNKI_TTNK-18600_CP011_Ingresar códig...	✓	25/05/2023 11:41	Documento de Mi...	230 KB
TUNKI_TTNK-18600_CP012_Ingresar a la p...	✓	25/05/2023 12:11	Documento de Mi...	258 KB
TUNKI_TTNK-18600_CP013_Presiona en e...	✓	25/05/2023 12:09	Documento de Mi...	294 KB
TUNKI_TTNK-18600_CP014_Ingresar corre...	✓	25/05/2023 12:10	Documento de Mi...	288 KB
TUNKI_TTNK-18600_CP015_Ingresar un co...	✓	25/05/2023 12:10	Documento de Mi...	318 KB
TUNKI_TTNK-18600_CP016_Ingresar un c...	✓	25/05/2023 12:10	Documento de Mi...	283 KB
TUNKI_TTNK-18600_CP017_Ingresar a la ...	✓	25/05/2023 14:55	Documento de Mi...	417 KB
TUNKI_TTNK-18600_CP018_Ingresar la OT...	✓	30/05/2023 09:44	Documento de Mi...	458 KB
TUNKI_TTNK-18600_CP019_Ingresar la OT...	✓	26/05/2023 15:34	Documento de Mi...	629 KB
TUNKI_TTNK-18600_CP020_Ingresar la O...	✓	25/05/2023 14:58	Documento de Mi...	509 KB
TUNKI_TTNK-18600_CP021_Ingresar códig...	✓	25/05/2023 14:59	Documento de Mi...	454 KB
TUNKI_TTNK-18600_CP022_Ingresar a la p...	✓	25/05/2023 15:48	Documento de Mi...	525 KB
TUNKI_TTNK-18600_CP023_Presiona en e...	✓	25/05/2023 16:03	Documento de Mi...	613 KB
TUNKI_TTNK-18600_CP024_No se ingresa...	✓	30/05/2023 10:50	Documento de Mi...	582 KB
TUNKI_TTNK-18600_CP025_No se ingresa...	✓	30/05/2023 10:50	Documento de Mi...	589 KB
TUNKI_TTNK-18600_CP026_No se ingresa...	✓	30/05/2023 10:50	Documento de Mi...	593 KB
TUNKI_TTNK-18600_CP027_Se completa ...	✓	25/05/2023 16:15	Documento de Mi...	640 KB

Figura 51. Evidencias de resultados de los sprint

## **Capítulo IV: APORTES A LA INSTITUCION**

Con el desarrollo de la experiencia profesional en el campo, para la mejora del aplicativo Tunki de la empresa financiera Interbank, se han logrado importantes aportes en el campo profesional y personal.

En el banco Interbank se han dado importantes aportes, ya que los diversos comentarios en la web sobre las limitaciones que tenía la aplicación desde la perspectiva de los clientes eran muy ácidos y hasta ofensivas por parte de estos clientes que no tenían una buena experiencia de usuario con el aplicativo tunki, en tal sentido la mejora de la experiencia de usuario (UX) con los cambios realizados por el equipo de desarrollo ha generado nuevos comentarios con mejor disposición de clientes hacia la aplicación, sin embargo se precisa que se está constantemente monitoreando las experiencias de estos usuarios para continuar en un ciclo de mejora continua 360°.

En cuanto a la empresa NTT DATA, como empresa líder en el mundo en provisión de servicios de alta tecnología, se tiene el principal aporte a la institución que es el posicionamiento de la empresa resultado de las buenas intervenciones realizadas en sus clientes estratégicos como es el Interbank. Estos resultados y aportes que la empresa obtiene, es el reflejo del equipo de conducción y desarrollo, que día a día gana mayor experiencia en el desarrollo profesional, lo que ha valido al área de tecnologías de información el reconocimiento por parte de la empresa.

Algunos conocimientos adquiridos en la universidad me han facilitado poder aportar como parte del equipo de mejoras del aplicativo Tunki, conocimiento sobre metodología como SCRUM, base de datos, Figma en diseño, el trabajo en equipo y la colaboración. Otros aportes en la experiencia relacionadas con el aseguramiento de la calidad han sido por formación personal y de capacitación de la empresa NTT DATA - Erevis y que se han aplicado en la experiencia.

## CONCLUSIONES

Con el aseguramiento de la calidad de las mejoras realizadas al aplicativo tunki del banco Interbank se ha llegado a las siguientes conclusiones:

1. En el desarrollo del presente proyecto, se han desarrollado capacidades de colaboración y trabajo en equipo para lograr el objetivo trazado por la empresa NTT DATA y la satisfacción del cliente.
2. La metodología aplicada SCRUM es ideal para que el equipo de trabajo se desarrolle sin mayor intervención de ninguna otra persona, por lo que el equipo ha aprendido a tomar diversas decisiones en el desarrollo del presente proyecto.
3. El constante control y monitoreo de los comentarios de los clientes es fundamental para encontrar sus necesidades, pero además retroalimentan al equipo constantemente.
4. En el proyecto se generan múltiples posibilidades de mejorar, que salen del mismo equipo de trabajo, por lo que se favorece el desarrollo profesional.
5. El rol de SCRUM MASTER es de vital importancia para lograr que el equipo adopte de manera natural el pensamiento ágil, sin el cual se hubiera presentado muchas discrepancias.

## RECOMENDACIONES

Los buenos resultados obtenidos con el proyecto, a pesar de ser satisfactorio para el cliente, estos por el contrario nos permiten seguir buscando nuevas oportunidades de mejora, por lo que se recomienda lo siguiente:

1. En la planificación del proyecto en el sprint review, es importante que se tenga la información de los comentarios de los clientes para conocer algunas posibilidades de mejorar la planificación e incluir o priorizar algunas historias de usuario (HU).
2. Para el diseño de debe continuar utilizando para los diversos proyectos la herramienta de FIGMA, que es más completa y eficiente que la anteriormente utilizada INVISIO.
3. Estar al pendiente del desarrollo tecnológico para que en las futuras intervenciones se puedan incorporar estas nuevas tendencias tecnológicas, como es el despegue que ha dado la inteligencia artificial y que se puedan incluir.

## REFERENCIAS BIBLIOGRAFICAS

- [1] A. R. Rendón Gutiérrez, «Automatización de pruebas para el aseguramiento de la calidad de Adminfo Vsmart», 2022, Accedido: 31 de marzo de 2023. [En línea]. Disponible en: <https://bibliotecadigital.udea.edu.co/handle/10495/30499>
- [2] S. Agarwal, S. K. Sonbhadra, y N. Singh Punn, «Software Testing and Quality Assurance for Data Intensive Applications», en *The International Conference on Evaluation and Assessment in Software Engineering 2022*, Gothenburg Sweden: ACM, jun. 2022, pp. 461-462. doi: 10.1145/3530019.3533678.
- [3] W. J. Arenas Saldaña, «Metodologías ágiles para mejorar la calidad del software en una entidad del estado, Lima 2021», *Repos. Inst. - UCV*, 2022, Accedido: 31 de marzo de 2023. [En línea]. Disponible en: <https://repositorio.ucv.edu.pe/handle/20.500.12692/85137>
- [4] G. Veloz-Remache, J. Menéndez-Verdecia, y L. Aguilar-Moncayo, «Mejores prácticas de calidad en el desarrollo de software integradas al conocimiento de la ingeniería», vol. 6, n.º 1, pp. 656-668, 2021, doi: 10.23857/pc.v6i1.2170.
- [5] C. A. Benites Luna, «Aplicación de aseguramiento de calidad de software y su influencia en proyectos de la empresa 3M, 2022», 2022, doi: 10.21142/tl.2022.2490.
- [6] F. Díaz Gil, «Metodologías de control de calidad y pruebas para soluciones tecnológicas basadas en AI/ML y Big Data», 2022, Accedido: 31 de marzo de 2023. [En línea]. Disponible en: <https://uvadoc.uva.es/handle/10324/57350>
- [7] J. A. G. García, C. R. de Verger-Ruano, y N. Sánchez-Gómez, «LA CALIDAD DEL SOFTWARE COMO MECANISMO DE ÉXITO EN PROYECTOS MULTIDISCIPLINARES: PROYECTO IMEDEA Y MEET2CARE».
- [8] J. Castaño-Henríquez y W. Castillo, «Métricas en la evaluación de la calidad del software : Una revisión conceptual Metrics in the evaluation of software quality : An Overview», vol. 2, n.º 2, pp. 21-26, 2021, doi: <http://doi.org/10.17981/cesta.02.02.2021.03>.
- [9] I. López Baldominos, «Aplicación de herramientas y análisis de datos para el aseguramiento de calidad del software», Universidad de Alcalá Escuela, 2020.
- [10] S. Amann y E. Jürgens, *The Future of Software Quality Assurance: Change-Driven Testing*. Alemania: Springer Open, 2020. Accedido: 26 de mayo de 2023. [En línea]. Disponible en:

<https://library.oapen.org/viewer/web/viewer.html?file=/bitstream/handle/20.500.12657/22847/1007314.pdf?sequence=1&isAllowed=y>

- [11] P. Orviz Fernández, M. David, D. C. Duma, E. Ronchieri, J. Gomes, y D. Salomoni, «Software Quality Assurance in INDIGO-DataCloud Project: a Converging Evolution of Software Engineering Practices to Support European Research e-Infrastructures», *J. Grid Comput.*, vol. 18, n.º 1, pp. 81-98, mar. 2020, doi: 10.1007/s10723-020-09509-z.
- [12] J. Singh y N. Bitew Kassie, «A study on software quality factors and metrics to enhance software quality assurance | International Journal of Productivity and Quality Management», vol. 29, n.º 1, pp. 24-44, 2020.
- [13] P. Temple, M. Acher, G. Perrouin, B. Biggio, J.-M. Jezequel, y F. Roli, «Towards Quality Assurance of Software Product Lines with Adversarial Configurations», en *Proceedings of the 23rd International Systems and Software Product Line Conference - Volume A*, Paris France: ACM, sep. 2019, pp. 277-288. doi: 10.1145/3336294.3336309.
- [14] Universidad de Minnessota. *Desarrollo de software Agile*, (2022). [En línea Video]. Disponible en: <https://www.coursera.org/learn/agile-software-development-es/lecture/SeSVJ/descripcion-general-de-scrum>
- [15] IONOS, «¿Qué es Jira? Todo lo que necesitas saber sobre el software de gestión de proyectos», IONOS Digital Guide. Accedido: 17 de octubre de 2023. [En línea]. Disponible en: <https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/que-es-jira/>
- [16] JUNTADEANDALUCIA, «TestLink y la Gestión de las pruebas | Marco de Desarrollo de la Junta de Andalucía». Accedido: 17 de octubre de 2023. [En línea]. Disponible en: <https://www.juntadeandalucia.es/servicios/madeja/contenido/recurso/378>

# ANEXOS

## Anexo 01: Equipo SCRUM y QA



Fig. 51. Equipo de desarrollo SCRUM, dev (front end y back end), QA y los arquitectos de software



Fig. 52. Reuniones diarias del equipo



## CONTRATO DE TRABAJO

Conste por el presente documento el Contrato de Trabajo a Plazo Indeterminado que celebran de una parte **Everis Perú S.A.C.** identificada con RUC N° 20521586134, con domicilio para estos efectos en Calle Dean Valdivia Nro. 148, San Isidro, Lima, debidamente representada por la Sra. Jannet Saravia Romero, identificada con Documento de Identidad N° 10207248, según nombramiento y facultades inscritos en la partida Nro. 12269845 del Registro de Personas Jurídicas de Lima, a quien en adelante se le denominará "EVERIS"; y de la otra parte, el (la) **Sr.(a) Quispe Garayar Leonela Alejandra**, identificado con Documento Nacional de Identidad N° 70659500, domiciliado en **Calle Buenos Aires Mz. S Lote 2 - Changuillo**, con correo electrónico **leoyale.1997.25@gmail.com** en adelante "Trabajador"; en los términos y condiciones contenidos en las cláusulas siguientes:

### PRIMERA: LAS PARTES

- 1.1. EVERIS es una persona jurídica dedicada, entre otras actividades, a la prestación de servicios de consultoría, asesoramiento y gestión empresarial; creación, diseño, desarrollo, implantación, explotación, reparación y mantenimiento de sistemas de información, infraestructuras tecnológicas y equipos informáticos, telemáticos, ofimáticos y automáticos; así como en la prestación de servicios de gestión y de externalización de toda clase de procesos de negocio, tanto a nivel nacional como internacional.
- 1.2. Por su parte, el Trabajador es una persona natural, quien luego de haber sido informado sobre las actividades de EVERIS y las funciones que EVERIS requiere que realice, declara estar capacitado para ocupar el puesto para el cual se le contrata.
- 1.3. Para efectos del ejercicio de su puesto, el Trabajador declara contar con las habilitaciones administrativas necesarias para el ejercicio del cargo.

### SEGUNDA: OBJETO

EVERIS contrata al Trabajador para que se desempeñe como **Centers Developer - CD** de la compañía. En el ejercicio de su cargo, el Trabajador deberá cumplir con todas las obligaciones inherentes a su posición y, a su vez, ejercitar las demás funciones que EVERIS le asigne, siguiendo las directrices emanadas del Gerente General, de sus superiores, o de cualquier otro representante de EVERIS.

Lo señalado es condición esencial para la suscripción del presente contrato y consecuente nacimiento de la relación laboral entre el Trabajador y EVERIS. Por ello, el incumplimiento del Trabajador de lo señalado o su negativa a acatar las disposiciones que se le comuniquen, constituirán una falta grave del Trabajador; la misma que otorgará a EVERIS el derecho de culminar la relación laboral.

El Trabajador declara además haber sido informado sobre que la asignación o reasignación de funciones o cargos o posiciones en EVERIS es potestad exclusiva de ésta, la misma que será ejercida dentro del marco legal vigente en el país.

Ambas partes convienen que el presente contrato se iniciará el día **15 del mes de marzo del año 2021**.

### TERCERA: OBLIGACIONES DEL TRABAJADOR

Son obligaciones del trabajador, entre otras, las siguientes:

- 3.1. Realizar en forma eficiente y oportuna las funciones que le sean asignadas.
- 3.2. Cumplir con lo estipulado en el presente contrato así como a las normas internas que EVERIS mantuviese vigente o que le comunicase a futuro.
- 3.3. Ejercer sus funciones en cualquiera de los establecimientos de EVERIS ubicados en la ciudad de Trujillo, según la compañía lo asigne a uno u otro.
- 3.4. Sin perjuicio de lo anterior, podrá desempeñarse también en otros lugares del país o del extranjero si así lo requieren las necesidades de EVERIS siempre que ello no implique un menoscabo para aquel.
- 3.5. Someterse a los exámenes médicos establecidos por Ley o convenidos con EVERIS, determinantes para la relación laboral o con la finalidad de cumplir las medidas higiénicas para evitar enfermedades o accidentes. La negativa del Trabajador a cumplir con lo señalado, constituye una causal de despido de conformidad con lo establecido en el artículo 23° del Texto Único Ordenado del Decreto Legislativo N° 728.
- 3.6. Asumir los costos en los que EVERIS incurra por dar cumplimiento al mandato judicial o administrativo de retención que pudiese iniciarse en contra del Trabajador.

considerará infructuoso el trato directo en caso tras transcurridos cinco días hábiles desde presentada la solicitud, las partes no hubieren llegado a un acuerdo.

#### **DÉCIMA OCTAVA: CLAUSULA RESOLUTORIA EXPRESA**

Las partes acuerdan que EVERIS podrá dar por terminada la relación laboral en forma unilateral, sin asumir por ello responsabilidad, en cualquier momento, en caso el trabajador incurra en cualquiera de los siguientes supuestos:

4. Incumpla con lo dispuesto en la Política Anticorrupción del Grupo Everis o cualquier otra norma interna, incluso comunicada con posterioridad a la suscripción de este documento, que regule las normas de prevención de la corrupción pública o privada.
5. No haber comunicado a EVERIS, por documento de fecha cierta, antes del inicio de la relación laboral, el haber sido sentenciado por la comisión de delito contra el patrimonio, delito contra la confianza y la buena fe en los negocios, delito contra los derechos intelectuales, delito contra la administración pública o delito contra la fe pública.
6. Hubiere desempeñado un cargo de confianza o dirección en alguna empresa inhabilitada para contratar con el Estado peruano, incluso si dicha inhabilitación ya hubiere cesado, sin haber notificado de ello a Everis por escrito de fecha cierta antes del inicio de la relación laboral.

Finalmente, el Trabajador declara haber leído y entendido esta cláusula, habiendo incluso contado con la disposición del empleador para explicársela, lo cual ha ocurrido. Siendo así, su contenido es claro y su suscripción obedece a la voluntad de ambas partes.

#### **DÉCIMANOVENA: NOTIFICACIONES Y DOMICILIO DE LAS PARTES**

Para todos los efectos del presente contrato, las partes señalan como sus respectivos domicilios los indicados en la introducción del presente documento, lugares a los cuales se tendrá por bien recibida cualquier comunicación entre ellas. Asimismo, el Trabajador autoriza a recibir notificaciones en el correo electrónico indicado en la introducción al presente documento, así como al correo electrónico corporativo que será proporcionado por EVERIS, teniéndolos por plenamente válidas.

Cualquier modificación del domicilio o del correo electrónico, para su validez, será efectiva después de los cinco días hábiles de haber sido comunicada mediante carta simple sobre papel, en el domicilio de EVERIS.

#### **VIGESIMA: LEY APLICABLE Y COMPETENCIA JURISDICCIONAL**

El presente contrato se rige por la legislación laboral peruana. Las partes acuerdan someterse a la jurisdicción exclusiva de Jueces y Tribunales del distrito judicial del Cercado de Lima, para dirimir cualquier controversia resultante o relacionada con el contrato, su ejecución, incumplimiento y/o terminación.

El presente contrato se extiende en dos ejemplares de idéntico tenor, quedando uno en poder del Trabajador y uno en poder de EVERIS.

Suscrito en la ciudad de Trujillo, Perú; el día **15 del mes de marzo del año 2021.**



SARAVIA ROMERO JANNET  
EVERIS PERU S.A.C.  
APODERADO



Quispe Garayar Leonela Alejandra  
TRABAJADOR

