



FACULTAD DE INGENIERÍA

Carrera de **INGENIERÍA DE SISTEMAS
COMPUTACIONALES**

“DESARROLLO DE UN APLICATIVO MÓVIL PARA
MEJORAR LA ATENCIÓN AL CLIENTE DE UNA
EMPRESA AUTOMOTRIZ DE LA CIUDAD DE
LIMA, 2022”

Tesis para optar al título profesional de:
Ingeniero de Sistemas Computacionales

Autor:

Stefano Ralf Tamayo Zegarra

Asesor:

Mg. Gabriel Augusto Tirado Mendoza
<https://orcid.org/0000-0001-8411-7684>

Lima - Perú

2023

JURADO EVALUADOR

Jurado 1 Presidente(a)	JORGE ALFREDO BOJÓRQUEZ SEGURA	10318709
	Nombre y Apellidos	Nº DNI

Jurado 2	MARITZA CABANA CACERES	40501601
	Nombre y Apellidos	Nº DNI

Jurado 3	EDUARDO REYES RODRIGUEZ	41212791
	Nombre y Apellidos	Nº DNI

INFORME DE SIMILITUD

Desarrollo de un aplicativo móvil para mejorar la atención al cliente de una empresa automotriz de la ciudad de Lima, 2022

INFORME DE ORIGINALIDAD



FUENTES PRIMARIAS

1	repositorio.uwiener.edu.pe Fuente de Internet	11%
2	hdl.handle.net Fuente de Internet	1%
3	docs.google.com Fuente de Internet	1%
4	podcast.unesp.br Fuente de Internet	1%

Excluir citas

Activo

Excluir coincidencias < 1%

Excluir bibliografía

Apagado

DEDICATORIA

Dedico este trabajo a Dios, por ser quien me da la fuerza y la sabiduría necesaria. A mis padres, por todo su esfuerzo y cariño. A mi tío Oscar, a mi tía Rocío y a mis primos. A mis 2 mejores amigas Karla y Tamara. A mis amigos, en especial, a un grupito que conocí hace poco y sin saberlo me han dado fuerzas para seguir. Y a una gran amiga, gracias quien los conocí.

AGRADECIMIENTO

Agradezco inmensamente a Dios, por ser mi soporte desde siempre. A mi tío Oscar que me ayudó con sugerencias y a contactar con los encuestados. Al asesor que con su paciencia y diligencia me ha guiado para el correcto desarrollo del presente trabajo investigativo.

Tabla de contenido

JURADO EVALUADOR	2
INFORME DE SIMILITUD	3
DEDICATORIA	4
AGRADECIMIENTO	5
TABLA DE CONTENIDO	6
ÍNDICE DE TABLAS	8
ÍNDICE DE FIGURAS	9
RESUMEN	10
ABSTRACT	11
CAPÍTULO I: INTRODUCCIÓN	12
1.1. Realidad problemática	12
1.2. Antecedentes	14
1.3. Bases teóricas	16
1.3.1. Variable dependiente	16
1.3.2. Variable independiente	18
1.4. Justificación	20
1.5. Formulación del problema	20
1.6. Objetivos	21
1.7. Hipótesis	22
CAPÍTULO II: METODOLOGÍA	23
2.1. Enfoque de la investigación	23
2.2. Alcance de la investigación	23
2.3. Tipo de investigación	24
2.4. Diseño de la investigación	24
2.5. Población y muestra	25
2.6. Técnica de recolección de datos	26
2.7. Instrumento de recolección o medición de datos	26
2.8. Instrumento de análisis de datos	27
2.9. Análisis de confiabilidad	28

2.10. Procedimiento de recolección de datos	28
2.11. Aspectos éticos	29
CAPÍTULO III: RESULTADOS	30
3.1. Aspectos técnicos	30
3.2. Hallazgos	58
3.2.1. Hallazgos de la primera hipótesis específica	59
3.2.2. Hallazgos de la segunda hipótesis específica	60
3.2.3. Hallazgos de la tercera hipótesis específica	62
3.2.4. Hallazgos de la cuarta hipótesis específica	63
3.2.5. Hallazgos de la hipótesis general	65
3.2.6. Prueba de normalidad	68
3.2.7. Prueba no paramétrica	69
3.3. Procedimientos tecnológicos	70
CAPÍTULO IV: DISCUSIÓN Y CONCLUSIONES	74
4.1. Discusión	74
4.2. Implicancias	76
4.3. Limitaciones	77
4.4. Conclusiones	78
4.5. Recomendaciones	79
REFERENCIAS	81
ANEXOS	83

Índice de tablas

Tabla 1. Estadísticas de fiabilidad	28
Tabla 2. Requerimientos identificados para el desarrollo del aplicativo móvil	35
Tabla 3. Análisis de frecuencia de la Pregunta 1 (accesibilidad a los datos) (pre-test).....	58
Tabla 4. Análisis de frecuencia de la Pregunta 1 (accesibilidad a los datos) (post-test).....	58
Tabla 5. Análisis de frecuencia de la Pregunta 2 (datos del vehículo) (pre-test)	59
Tabla 6. Análisis de frecuencia de la Pregunta 2 (datos del vehículo) (post-test).....	60
Tabla 7. Análisis de frecuencia de la Pregunta 3 (datos de servicios) (pre-test).....	61
Tabla 8. Análisis de frecuencia de la Pregunta 3 (datos de servicios) (post-test)	61
Tabla 9. Análisis de frecuencia de la Pregunta 4 (datos de repuestos) (pre-test).....	62
Tabla 10. Análisis de frecuencia de la Pregunta 4 (datos de repuestos) (post-test)	63
Tabla 11. Análisis de frecuencia de la Pregunta 5 (registrar) (pre-test)	64
Tabla 12. Análisis de frecuencia de la Pregunta 5 (registrar) (post-test)	64
Tabla 13. Promedio de tiempo por cada pregunta (pre-test)	65
Tabla 14. Promedio de tiempo por cada pregunta (post-test).....	66
Tabla 15. Promedio de tiempo por cada colaborador (pre-test)	67
Tabla 16. Promedio de tiempo por cada colaborador (post-test).....	68
Tabla 17. Pruebas de normalidad	68
Tabla 18. Prueba de rangos con signo de Wilcoxon.....	69
Tabla 19. Prueba estadística de Wilcoxon.....	70
Tabla 20. Recursos tecnológicos necesarios para la investigación	71
Tabla 21. Conocimientos tecnológicos necesarios para la investigación	72
Tabla 22. Roles de Scrum en el proyecto	73

Índice de figuras

Figura 1. Proceso de atención al cliente AsIs	31
Figura 2. Proceso de atención al cliente ToBe	32
Figura 3. Prototipo de Iniciar sesión.....	36
Figura 4. Prototipo de Iniciar sesión con Google	36
Figura 5. Prototipo de Registrarse	37
Figura 6. Prototipo de Inicio.....	37
Figura 7. Prototipo de Consultar datos del vehículo	38
Figura 8. Prototipo de Consultar repuestos	38
Figura 9. Prototipo de Agregar repuesto	39
Figura 10. Prototipo de Consultar Servicios.....	39
Figura 11. Prototipo de Agregar servicios.....	40
Figura 12. Prototipo de Generar QR.....	40
Figura 13. Prototipo de Escanear QR	41
Figura 14. Modelo lógico de la base de datos	42
Figura 15. Modelo físico de la base de datos	43
Figura 16. Entidad Categoría.....	44
Figura 17. Entidad Repuesto	44
Figura 18. Entidad Repuesto Comprado.....	45
Figura 19. Entidad Servicio	45
Figura 20. Entidad Servicio Comprado	46
Figura 21. Entidad Vehículo.....	46
Figura 22. Captura de front-end de Inicio de sesión	47
Figura 23. Captura de front-end de Registro	47
Figura 24. Captura de front-end de Home.....	48
Figura 25. Captura de front-end de Consultar placa.....	48
Figura 26. Captura de front-end de Consultar repuestos o servicios.....	49
Figura 27. Captura de front-end de Agregar repuesto o servicio	49
Figura 28. Captura de front-end de Generar QR	50
Figura 29. Captura de front-end de Escanear QR.....	50
Figura 30. Prueba en dispositivo físico de Home.....	51
Figura 31. Prueba en dispositivo físico de Consultar placa.....	52
Figura 32. Prueba en dispositivo físico de Consultar repuestos	52
Figura 33. Prueba en dispositivo físico de Agregar repuesto	53
Figura 34. Prueba en dispositivo físico de Consultar servicio	53
Figura 35. Prueba en dispositivo físico de Agregar servicio	54
Figura 36. Prueba en dispositivo físico de Generar QR	54
Figura 37. Prueba en dispositivo físico de Escanear QR.....	55
Figura 38. Prueba en dispositivo físico de Escáner funcionando.....	55
Figura 39. Captura de selección de dispositivos a probar	56
Figura 40. Resultados de las pruebas en Firebase	56
Figura 41. Comparación del promedio de tiempos entre pre-test y post-test.....	67

RESUMEN

Muchas empresas del rubro automotriz suelen enforzar su tecnología a procesos internos y no le dan mucha importancia a la atención al cliente. El trabajo de investigación presentado tuvo como objetivo desarrollar una aplicación móvil para mejorar la atención al cliente en una empresa del rubro automotriz de la ciudad de Lima, 2022. Para ello, se siguieron una serie de pasos para el desarrollo de un aplicativo móvil que solucione la problemática planteada.

Se realizó una investigación de tipo aplicada con un diseño pre y post test. La población estuvo conformada por 8 colaboradores de una empresa automotriz. Para realizar la recolección de datos, la técnica aplicada fue la encuesta, cuyo instrumento fue un cuestionario digital. Los datos obtenidos fueron analizados por medio de la prueba estadística de Wilcoxon con ayuda del software SPSS.

Los resultados evidencian que existe una disminución de entre el 30% y 40% en el tiempo de atención al cliente, en las diferentes etapas de la atención. En base a los hallazgos de esta investigación, se concluye que el aplicativo móvil afectó de forma positiva a la atención al cliente, reduciendo los tiempos. Adicionalmente, se mejoró el proceso de atención mediante la reducción de tiempos y eliminación de actividades innecesarias

PALABRAS CLAVES: atención al cliente, aplicativo móvil, reducción de tiempos, mejora de proceso, rubro automotriz.

ABSTRACT

A great number of automotive companies are used to focus their technological resources to internal processes, and they don't place much value on customer support. The research project had as objective developing a mobile application to enhance customer support in an automotive company from de city of Lima, 2022. Hence, a series of steps were followed for the development of a mobile application that revolves the problem raised.

Applied research was executed, with a design pre-test / post-test. Population was form by 8 collaborators of an automotive company. For data recollection, the technique applied was a survey, which instrument was a digital questionnaire. The obtained data were analyzed by Wilcoxon's Statistic Test helped by SPSS Software.

The results prove that there is a diminishment between 30% and 40% in customer support's time and in its different stages. Based on the discoveries of this research, it's concluded that the mobile application affected customer support in a positive way, by reducing times. Moreover, the customer support process was improved through time reduction and unnecessary activities deletion.

Keywords: customer support, mobile application, time reduction, process improvement, automotive industry.

CAPÍTULO I: INTRODUCCIÓN

1.1. Realidad problemática

Actualmente, las empresas internacionales del rubro automotriz aplican tecnología en su rubro, esto les ayuda a gestionar distintos procesos. Además, se aplica la tecnología para su apropiada capacitación, en distintos centros de estudio especializados. La tecnología se aplica en distintos ámbitos incluso algunas empresas la usan para la atención al cliente, muchos de ellos incluso sin que el cliente salga de su domicilio o fungiendo como herramienta para las personas que están a cargo de la atención al cliente y así desempeñar mejor su función. En el estudio de (Neri Guzmán, 2022) se registraron tecnologías y equipamiento usando un formato de registro para cada una de las tecnologías disponibles. Además de ello, dicho formato también registró cierta información sobre el personal que es el encargado de operar dichas tecnologías, año de adquisición e intensidad de uso. En el diseño y la recolección participaron ingenieros los cuales cuenta con experiencia en la industria automotriz.

En el ámbito nacional, la tecnología se aplica en menor medida, dependiendo del sector del país en el que se encuentre la empresa del rubro automotriz o el reconocimiento de esta. En aquellas empresas en las que se aplica, esta tecnología está enfocada principalmente a los procesos internos de la industria. También existen algunas empresas más grandes, las cuales usan la tecnología o los aplicativos para comunicación. Además, existe un gran sector que no accede a estas soluciones tecnológica por motivos económicos; por otro lado, algunas empresas tienen los medios económicos para financiar soluciones tecnológicas, pero no saben cómo aplicar correctamente dichas tecnologías o solo las aplican a un área. Aldo Canales (Diario Gestión, 2014, p. 11) menciona: “Es

necesario que las empresas se adapten a las nuevas tecnologías para que sobrevivan en el mercado. En años pasados, comenzamos con las tendencias de las PCs, hoy nos encontramos en una ola de móviles, por lo que es necesario contar con plataformas propias, aplicaciones personalizadas para servicios y productos cada vez más personalizados”.

Localmente, también se aplica la tecnología, pero no está enfocada en la atención al cliente. Debido a la naturaleza del rubro automotriz, las empresas pertenecientes a este rubro cuentan con amplias instalaciones y solo cuentan con un centro de datos. Los clientes deben esperar a que la persona que los atiende vaya hasta el centro de datos para consultar su estado, si es un cliente registrado, los repuestos comprados, etc. Esto resulta en la demora en la atención al cliente y por tanto una experiencia deficiente. En estos lugares se aplica la tecnología, pero no de la forma en que deberían o solo la enfocan en algún área, sin considerar que la tecnología es muy amplia y podría aplicarse a diferentes áreas, como la atención al cliente. En esta área de atención al cliente se podría aplicar la tecnología por medio de un aplicativo móvil, el cual facilite a los colaboradores de la empresa la gestión de los datos, lo cual resulte en una mejora significativa en la atención al cliente. En el mismo artículo de (Diario Gestión, 2014) Aldo Canales también explica que las aplicaciones móviles ayudan a reducir costos a las empresas. Además, según cifras del Gremio de Tecnologías de la Información de la Cámara de Comercio de Lima, el Perú ya ha iniciado su proceso de adaptación invirtiendo el 9.9% de su inversión en tecnología en adquirir e implementar software en el sector público y privado.

1.2. Antecedentes

El uso de aplicativos móviles ya ha demostrado ser una herramienta muy eficaz para mejorar los procesos de atención al cliente, en distintos entornos empresariales. Con anterioridad se ha usado esta solución obteniendo resultados satisfactorios, tras la implementación de una solución de este tipo.

(Panduro Lara & Tello Pacheco, 2020) proponen en su tesis la reducción de tiempo en los procesos de entrada y salida del personal mediante el uso de un aplicativo móvil. Se usó una ficha de registro y se apoyaron de la observación. Usaron la correlación de Pearson para el análisis de sus datos. Finalmente, su hipótesis fue validada y se concluyó que existe una correlación entre la implementación del aplicativo móvil y el tiempo que consumen la entrada y salida de los colaboradores.

En la tesis de (Crisostomo Aylas, 2020) se realizó la implementación de una aplicación móvil para mejorar la eficiencia en la reducción del tiempo de recepción de un centro de distribución. Se usaron las fichas de registro de información como instrumento de recolección de datos. En esta investigación se usó la correlación de Pearson pre-test y post-test. Tras analizar los datos obtenidos, se comprobó que existe una reducción en las horas hombre, al realizar la implementación de la aplicación móvil en el Centro de Distribución.

(Huaman Cruz, 2020) explica en su artículo que la investigación realizada por él, le permite afirmar que el uso de las aplicaciones móviles crece exponencialmente y está incursionando rápidamente en diferentes rubros, es solo cuestión de tiempo para su completa expansión. Además, detalla que el usuario ve con buenos ojos el uso de estos aplicativos ya que se ven mejoras en el alcance y prioriza que sus necesidades sean

cubiertas, además de tener un gran ambiente de desarrollo y apuntar a crecer masiva y diariamente con el usuario final.

En el mismo artículo publicado por (Huaman Cruz, 2020) se comprueba la existencia de un impacto creciente y positivo en la atención al cliente. Se demostró la relación que tiene con los aplicativos móviles en sus diferentes etapas y en diferentes sectores como: transporte, salud, arte, educación e industria agrícola. Todo esto basado en el conocimiento de las diversas necesidades del consumidor con la finalidad de cubrir las necesidades actuales y sus requerimientos y necesidades futuras, para así obtener resultados óptimos. Considerando estas características, el autor señala que se podrá generar un cambio de cultura organizacional, mejorando así la calidad del servicio.

En la tesis de (Vergara Vásquez, 2019) se concluyó la existencia de la relación entre la aplicación móvil y la atención al cliente, teniendo una alta relación (65%). El aplicativo recibió una aceptación directa por parte de la empresa. Mejorando distintos procesos que manejaba la empresa como la llegada al cliente, la preparación de los platos del restaurante y el proceso de entrega y pago. Esto también permitió detectar los cuellos de botella que existe en los diferentes procesos del restaurante. Todos estos procesos fueron mejorados y agilizados, gracias al uso del aplicativo móvil.

En un artículo procedente de Arabia Saudita, publicado en ingles por (Khrais & Alghamdi, 2021) se detalla que, con la llegada creciente del desarrollo tecnológico, los usuarios prefieren la conveniencia al acceder a los servicios desde plataformas digitales, en otras palabras, facilitar la gestión de los procesos para acceder al servicio de una forma más oportuna. Señala también que el alto nivel de aceptación de las aplicaciones móviles entre los usuarios al rededor del mundo refleja los beneficios convincentes, como: una

amplia escala de servicios, mayor flexibilidad y transacciones más rápidas. El autor menciona algo muy importante, que las aplicaciones móviles son requeridas para sobrevivir a las consecuencias de la globalización y los mercados abiertos, esto funciona gracias a la personalización de los servicios. En los próximos años el crecimiento y rentabilidad de las empresas aumentará dependiendo de las habilidades que tengan para adaptarse a las necesidades de sus clientes, solo podrán mantenerse en competencia de esta forma y adaptando sus modelos de negocio para asegurar que a los clientes se les ofrezca una experiencia consistentemente personalizada.

1.3. Bases teóricas

1.3.1. Variable dependiente

La atención al cliente es el medio por el cual las empresas pueden crecer y permanecer en el mercado, todo esto debido a la fidelidad de sus clientes, ya que, sin ellos no habría a quien ofrecerles sus servicios, en otras palabras, su rentabilidad sería nula y no cumplirían con sus objetivos. Por lo anterior expuesto se debe reflexionar sobre la gran relevancia que tienen los clientes en el funcionamiento de las organizaciones. (Duque Fierro, Saldarriaga Castrillón, & Bohorquez Ordoñez, 2016).

El artículo de (Morales Reyna, Terranova Borja, & Sandoya Mayorga, 2019) nos aclara un punto muy importante, el cual es la diferencia entre la atención al cliente y el servicio al cliente. El servicio al cliente entiende como el conjunto de procesos que una empresa le ofrece al cliente. Por otro lado, la atención al cliente se define como el trato que tienen los colaboradores de la empresa con el cliente. Manteniendo siempre en perspectiva que el cliente es la parte más importante de la empresa. Sin embargo, ambos van de la mano, ya que así el cliente reciba el mejor trato, si no consigue el servicio por el que pagó,

este no estará satisfecho.

En el artículo de (Valenzuela Salazar, Buentello Martínez, Alanis Gomez, & Villareal Sánchez, 2019) también se resalta la importancia que está tomando la atención al cliente en los últimos años, no solo en un rubro en particular sino dentro de todo proceso de negocio, esto debido a las crecientes expectativas de los clientes y la competitividad en el mercado, ya que toda empresa busca igualar o superar dichas expectativas. Por ello se deben ofrecer buenos servicios, anticipándose a responder una amplia cantidad de posibles necesidades de los clientes. Finalmente, se señala que las empresas en el futuro deberán esforzarse mucho más si quieren mantenerse en el mercado, ya que como presenciamos día a día se vive una competencia agresiva en todos los rubros para poder captar una mayor cantidad de clientes. La satisfacción o anticipación de sus necesidades resultará en la lealtad de los clientes y buena reputación de la empresa.

El artículo publicado en ingles por (Song, Liu, Guo, Yang, & Jin, 2022) ,el cual se ubica en un contexto postpandemia, provee nuevas variables explicativas las cuales influyen en los factores de satisfacción. Entre estas variables están los temas y los sentimientos, los cuales deben de ser considerados por las diferentes empresas. Además, su estudio enciende una luz en el vínculo entre los temas de experiencia de los clientes y su satisfacción, sugiriendo que User-Generated Content (UGC), como opiniones online, pueden ser usados para identificar estructuras significativas entre aspectos y atributos relacionados al servicio que ofrece la empresa.

Actualmente, la sociedad se enfrenta a algunos problemas con respecto a la atención al cliente, lo cual queda evidenciado en el artículo de (Enrique Villegas, Meibol Fajardo, & Vásquez, 2014) . En su estudio se demuestra que los clientes en su rubro de investigación

no están siendo atendidos de una forma óptima, lo cual produce que se generen quejas constantes provenientes de los clientes del área comercial, lo cual refleja una gestión deficiente en las empresas de dicho rubro. Insiste en que se debe incorporar como parte de la atención al cliente las mediciones periódicas sobre como los clientes perciben la atención, esto en aras de lograr mejoras en el servicio.

En el artículo de (Duque Fierro, Saldarriaga Castrillón, & Bohorquez Ordoñez, 2016) se señala que las organizaciones deben tener su atención puesta en cada gesto o señal que los clientes les brinden para poder conocer y estudiar sus necesidades y proveerles el servicio exacto que necesitan, sobre todo en un mercado tan competitivo, como es el actual. La observación y el conocimiento del cliente aporta mucho también para solucionar distintas problemáticas que puedan presentarse, analizándolas y eligiendo la mejor manera de solucionarlas.

1.3.2. Variable independiente

El concepto de aplicaciones móviles está definido como aplicaciones de software que son ejecutadas en smartphones o tabletas y son distribuidas por medio de servicios como iTunes (para iPhone y iPads) o Google Play (para aplicaciones Android). Estas aplicaciones pueden ser de autoría de desarrolladores u otros individuos u organizaciones, este tipo de aplicaciones son llamadas aplicaciones de terceros. (van-Velsen, JMA-Beaujean, & van-Gemert-Pijnen, 2013).

En el artículo de (Gasca Mantilla, Camargo Ariza, & Medina Delgado, 2014) se detallan algunos puntos importantes sobre el desarrollo de aplicativos móviles. Las aplicaciones móviles deben tener en consideración las características del entorno de ejecución, es decir las características del teléfono, para que se pueda garantizar su

apropiado funcionamiento. Señala que las aplicaciones móviles son muy útiles para solventar tanto problemas generales como particulares, esto gracias a sus características de movilidad y ubicuidad. Se indica que los entornos integrados de desarrollo (IDE) nos proveen de herramientas de edición, compilación, depuración, análisis y ejecución en un mismo lugar, permitiéndonos mejorar los tiempos y facilitar el desarrollo. Finalmente, el artículo brinda una información muy importante. Android permite crear formatos de interfaz más amigables y sin una gran complejidad, debido a la edición mediante archivos XML.

En el artículo de (Monsalve Sossa, Zapata, Gómez Adasme, & Ríos, 2018) , se concluye que debido a la incursión de los aplicativos móviles, es necesario ampliar las habilidades de pruebas para garantizar la calidad del aplicativo en los distintos dispositivos móviles. Dichas habilidades deben centrarse en probar la aplicación desde distintas dimensiones. Actualmente, es muy difícil realizar estas pruebas debido a la cantidad combinatoria de las variables a probar, como: sistemas operativos, versión, tamaño, fabricante, etc. Por tanto, se sugiere realizar estas pruebas reduciendo los costos y aumentando la cobertura del proyecto.

En el artículo de (Dascalu, y otros, 2018) se puede determinar que los aplicativos móviles son muy necesarios en muchos y diversos rubros. Implementar el aplicativo móvil en Android nos brinda muchas ventajas como la cantidad de personas que tendrán acceso. Además, se recomienda el uso de Cohen Williamson Questionnaire para determinar el nivel de estrés que soportaría nuestro aplicativo móvil.

(Santamaría Puerto & Hernández Rincón, 2015, p. 4) explican en su artículo que la aparición de nuevas tecnologías, tanto como su expansión, desarrollo y creación son

capaces de modificar nuestras vidas, sin permitirnos ser ajenos a dichos cambios y aportan potenciales beneficios para mejoras. También, señala que las aplicaciones móviles han crecido enormemente y que su uso es creciente, particularmente en el rubro del estudio y que está formando parte de la cotidianidad.

1.4. Justificación

En esta investigación, se busca desarrollar un aplicativo móvil para mejorar la atención al cliente en una empresa del rubro automotriz. Esto es debido a que todo proceso de negocio inicia y termina con un mismo sujeto, el cliente. Debido a ello el cliente es la parte más importante de toda empresa, por ello es crucial mejorar la atención al cliente. Sin embargo, investigaciones referidas a la mejora de procesos hacia el cliente en el rubro automotriz, no ha sido abordada completamente, debido a la naturaleza de las empresas de este rubro. Es por ello, que esta investigación aporta a la mejora del proceso de atención al cliente en el rubro automotriz, respecto a la reducción de tiempos.

1.5. Formulación del problema

Tras la realidad problemática expuesta, se formula la siguiente pregunta: ¿De qué manera el desarrollo de un aplicativo móvil mejora la atención al cliente de una empresa automotriz de la ciudad de Lima, 2022?

En base a ello se obtuvieron los siguientes problemas específicos:

- ¿Cómo un aplicativo móvil puede ayudar en la agilización del proceso de atención al cliente?
- ¿De qué manera mejora la experiencia del cliente usando un aplicativo móvil como apoyo tecnológico a quienes les brindan atención?

1.6. Objetivos

Esta investigación tiene como objetivo general: Desarrollar una aplicación móvil para mejorar la atención al cliente en una empresa del rubro automotriz de la ciudad de Lima, 2022. Para poder alcanzar el objetivo general anteriormente expuesto, primero se deben completar los siguientes objetivos específicos:

- Desarrollar una aplicación móvil para mejorar el tiempo de obtención de datos de un vehículo, en una empresa automotriz.
- Desarrollar una aplicación móvil para mejorar el tiempo de obtención de datos de los servicios aplicados, en una empresa automotriz.
- Desarrollar una aplicación móvil para mejorar el tiempo de obtención de datos de los repuestos instalados, en una empresa automotriz.
- Desarrollar una aplicación móvil para mejorar el tiempo de registro de datos de los repuestos y servicios, en una empresa automotriz.

La dimensión usada para la variable dependiente fue el tiempo de atención, la cual cuenta con los siguientes indicadores:

- Tiempo de obtención de datos de un vehículo
- Tiempo de obtención de datos de los servicios aplicados
- Tiempo de obtención de datos de los repuestos instalados
- Tiempo de registro de datos de los repuestos y servicios.

1.7. Hipótesis

Basándose en lo expuesto con anterioridad, la hipótesis formulada es la siguiente: La aplicación móvil mejora la atención al cliente en una empresa del rubro automotriz de la ciudad de Lima, 2022. Además, para probar la hipótesis se plantearon las siguientes hipótesis específicas:

- La aplicación móvil mejora el tiempo de obtención de datos de un vehículo, en una empresa automotriz.
- La aplicación móvil mejora el tiempo de obtención de datos de los servicios aplicados, en una empresa automotriz.
- La aplicación móvil mejora el tiempo de obtención de datos de los repuestos instalados, en una empresa automotriz.
- La aplicación móvil mejora el tiempo de registro de datos de los repuestos y servicios, en una empresa automotriz.

CAPÍTULO II: METODOLOGÍA

2.1. Enfoque de la investigación

(Hernández Sampieri, Fernández Collado, & Baptista Lucio, 2014, p. 4) define: “El enfoque cuantitativo es secuencial y probatorio. Parte de una idea que va acotándose y, una vez delimitada, se derivan objetivos y preguntas de investigación, se revisa la literatura y se construye un marco o una perspectiva teórica. De las preguntas se establecen hipótesis y determinan variables; se traza un plan para probarlas (diseño); se miden las variables en un determinado contexto; se analizan las mediciones obtenidas utilizando métodos estadísticos, y se extrae una serie de conclusiones respecto a la o las hipótesis.”

La investigación tuvo un enfoque cuantitativo, debido a que contó con una pregunta de investigación, objetivos, un marco teórico y una hipótesis con variables, que fueron probada después de realizar las mediciones correspondientes. Finalmente, se llegó a una conclusión.

2.2. Alcance de la investigación

En el libro de (Hernández Sampieri, Fernández Collado, & Baptista Lucio, 2014, p. 95) se detalla que los estudios explicativos no se limitan a la descripción de conceptos o fenómenos o al establecimiento de relaciones entre estos. Se dirigen a brindar una respuesta a las causas de determinados eventos y fenómenos físicos o sociales. Como su nombre deja en evidencia, está enfocado en explicar por qué ocurre un fenómeno y en qué condiciones se manifiesta su relación entre dos o más variables.

Debido al concepto expuesto con anterioridad, se pudo fundamentar que el alcance

de esta investigación fue de tipo explicativo, ya que estuvo orientada a la comparación de 2 variables, para hallar la relación que tienen entre sí. En este caso las variables fueron el aplicativo móvil que se desarrollará y la atención al cliente.

2.3. Tipo de investigación

Según (Baena Paz, 2014, p. 22) el objetivo de la investigación aplicada está destinado específicamente a la acción. Este tipo de investigación puede aportar hechos nuevos, si proyectamos suficientemente bien nuestra investigación, para que así se pueda confiar en los hechos puestos al descubierto. La nueva información puede ser útil y estimable para la teoría.

En base a lo expuesto por (Baena Paz, 2014), la investigación fue de tipo aplicada ya que su propósito es resolver el problema relacionado a la atención del cliente de una empresa automotriz mediante el desarrollo de una aplicación móvil.

2.4. Diseño de la investigación

(Baena Paz, 2014) explica en su libro, que el método experimental es el procedimiento científico, el cual permite la inducción de las relaciones empíricas entre variables con el fin de comprobar una hipótesis, por medio de un experimento controlado.

En concordancia con lo expuesto por (Baena Paz, 2014), se pudo definir que la investigación tuvo un método experimental, debido a que se realizó un proceso de experimentación para validar la veracidad de la hipótesis planteada. Se enfocó de forma preexperimental, realizando las mediciones correspondientes antes y después de la implementación.

(Hernández Sampieri, Fernández Collado, & Baptista Lucio, 2014) explica que en el diseño pre-test / post-test con un solo grupo, se le aplica una prueba previa al estímulo experimental, luego se administra el estímulo y finalmente se aplica una prueba posterior.

Se diagramaría de la siguiente manera:

G: O₁ X O₂

Siendo:

O₁: Información recopilada de la variable dependiente antes de la implementación de la variable independiente.

X: Implementación de la variable independiente

O₂: Información recopilada de la variable dependiente después de la implementación de la variable independiente.

De acuerdo con el concepto explicado anteriormente, la investigación usó un diseño preexperimental, pre-test / post-test con un solo grupo, de tal forma que se hizo una prueba de la situación previa a la solución y tras implementar la solución, se hizo otra prueba. Esto nos ayudó a tener un punto de referencia.

2.5. Población y muestra

De acuerdo con (Cruz, Olivares, & González, 2014) la población está conformada por todos los miembros de un grupo determinado. Normalmente, es costoso y requiere de considerable inversión de tiempo la evaluación de toda la población de interés.

Considerando lo mencionado anteriormente, se estableció que la población fueran 8

colaboradores que están encargados directamente de los procesos de atención al cliente, en una empresa del rubro automotriz de la ciudad de Lima, lo cual permite, a través de los datos obtenidos, realizar la validación de las hipótesis.

(Hernández Sampieri, Fernández Collado, & Baptista Lucio, 2014, p. 300) señalan que una investigación no siempre se cuenta con una muestra, sino que en la mayoría de los casos se suele usar una muestra para economizar tiempos y recursos.

En base a lo expuesto anteriormente, dado que la población de esta investigación son 8 personas, se determinó que no afecta en la economía de tiempos y recursos. Debido a ello, se señala que en esta investigación se está considerando la totalidad de la población.

2.6. Técnica de recolección de datos

(Hernández Sampieri, Fernández Collado, & Baptista Lucio, 2014, p. 159) citando a (Archeater, 2005), se señala que: Las encuestas de opinión son consideradas por muchos autores como un diseño o método para investigaciones no experimentales transversales o transeccionales descriptivas o correlacionales-causales, ya que a veces tienen los propósitos de unos u otros diseños y a veces de ambos.

Por lo mencionado anteriormente, la técnica que se usó fue la encuesta, ya que puede ser aplicada en diferentes diseños de investigación, incluyendo el diseño experimental, el cual fue el que se aplicó en la investigación. Además, se consideró como la más adecuada para el análisis cuantitativo.

2.7. Instrumento de recolección o medición de datos

(Hernández Sampieri, Fernández Collado, & Baptista Lucio, 2014) citando a

(Chasteauneuf, 2009) se explica que: En fenómenos sociales, el instrumento más utilizado para la recolección de datos es el cuestionario y se usa en encuestas de todo tipo. Un cuestionario consiste en un conjunto de preguntas respecto a una o más variables a medir.

Considerando lo anteriormente expuesto, el instrumento seleccionado para la recolección de datos fue el cuestionario, el cual es muy flexible y ha sido probado como un instrumento óptimo, además de ser el más utilizado al momento de realizar la investigación. Se creó el cuestionario haciendo uso de Google Forms, el cual fue adjuntado en el **Anexo N°3**. Se obtuvo la validez del instrumento por medio de 2 juicios de expertos detallados en los **Anexo N°4 y 5**. El instrumento fue validado por el docente de la UPN, Guido Trujillo Valdiviezo, quien es ingeniero de sistemas, profesor universitario de posgrado y pregrado de Metodología de la investigación y cuenta con un diplomado en Métodos y Técnicas de Investigación. Además, está calificado como investigador Renacyt. También se obtuvo la validez por parte de Oscar Emiliano Grados Meza, quien es ingeniero industrial, técnico en Automotores Diesel y tiene diplomatura en Mantenimiento y Confiabilidad en la PUCP.

2.8. Instrumento de análisis de datos

De acuerdo con (Hernández Sampieri, Fernández Collado, & Baptista Lucio, 2014), tras codificar los datos, transferirlos a una matriz, guardarlos en un archivo y limpiar los errores, se procede a realizar el análisis de los datos.

Para realizar el procedimiento de análisis de datos se usó el software SPSS. (Ferrán, 2001) define que SPSS es un sistema amplio y flexible para el análisis estadístico y gestión de datos en un entorno gráfico, se maneja mediante menús descriptivos y cuadros de

diálogos. Por lo que se considera que es el instrumento de análisis adecuado para esta investigación.

2.9. Análisis de confiabilidad

Se hizo uso del Alfa de Cronbach para medir la confiabilidad del instrumento. Para calcular el alfa de Cronbach se usó el software SPSS versión 28. Este análisis de confiabilidad fue aplicado con base a las respuestas pertenecientes al post test.

Tabla 1.

Estadísticas de fiabilidad

Alfa de Cronbach	N de elementos
.926	5

Fuente: Elaboración propia

El Alfa de Cronbach obtenido es 0.926. Debido a ello, podemos decir que el instrumento usado tiene un alto coeficiente de confiabilidad.

2.10. Procedimiento de recolección de datos

El procedimiento de recolección de datos constó de ciertos pasos. Primero, identificar el problema con relación a la atención al cliente. Luego, crear un cuestionario virtual en Google Forms con preguntas cerradas con el fin de medir que tan bueno es el proceso de atención al cliente. El cuestionario fue auto aplicado e individual. A continuación, se envió el cuestionario a cada participante por medios digitales, indicándoles el motivo de la encuesta y como serían usadas sus respuestas. Los participantes son aquellos que se encuentran en la muestra, la cual en esta investigación fue

la totalidad de la población, es decir, 8 encuestados. A continuación, se realizó el desarrollo del aplicativo móvil para poder realizar una mejora en los tiempos de atención al cliente. El aplicativo fue mostrado a los usuarios y se les brindó el acceso al aplicativo. Finalmente, se volvió a aplicar el mismo cuestionario, con el fin de evaluar sus respuestas después de haber usado el aplicativo móvil, y se procedió con el análisis de los datos obtenidos.

2.11. Aspectos éticos

Con respecto a los aspectos éticos, la investigación ha sido orientada en base a las líneas y sub líneas de investigación brindadas por la universidad. Además, se realizaron las referencias bibliográficas correspondientes a cada autor, de acuerdo con la normativa vigente de la universidad, las cuales son las normas APA. El proceso de recolección de información y los datos resultantes fueron reales. Cabe recalcar que las personas encuestadas han sido comunicadas propiamente sobre el motivo de la encuesta y el uso que tendrían sus respuestas. Se obtuvo el consentimiento informado, el cual se describe en la descripción del formulario (Anexo N°3), señalando como se usarían sus respuestas. Además, se usó una codificación de los datos utilizando el termino colaborador y asignándole un numero a cada uno con la finalidad de proteger los datos personales de los participantes en la encuesta. Finalmente, la información brindada por los encuestados ha sido tratada con estrictas medidas de confidencialidad.

CAPÍTULO III: RESULTADOS

3.1. Aspectos técnicos

Para la investigación se identificó el objetivo general y para alcanzar dichos objetivos, se tuvieron que lograr los objetivos específicos basados en la medición de las dimensiones de la variable dependiente.

Previo a ello se realizó el desarrollo del aplicativo móvil y se recolectaron las opiniones, los pasos que se siguieron para la elaboración fueron:

- Analizar requerimientos del aplicativo móvil
- Preparar prototipos previos al desarrollo del aplicativo móvil
- Modelar la base de datos del aplicativo móvil
- Crear las tablas o entidades que conforman la Base de Datos
- Desarrollar el front-end del aplicativo móvil
- Desarrollar el back-end del aplicativo móvil
- Realizar las pruebas, para validar las funcionalidades del aplicativo móvil
- Presentar el aplicativo móvil a los usuarios
- Recolectar opiniones sobre el uso del aplicativo móvil

A continuación, mostramos el proceso AsIs y ToBe del proceso de atención al cliente, cuya mejora fue el objetivo de la investigación. El AsIs es el proceso inicial de cómo se realizaba la atención al cliente. Por otro lado, el ToBe es el proceso mejorado gracias al aplicativo móvil desarrollado en esta investigación.

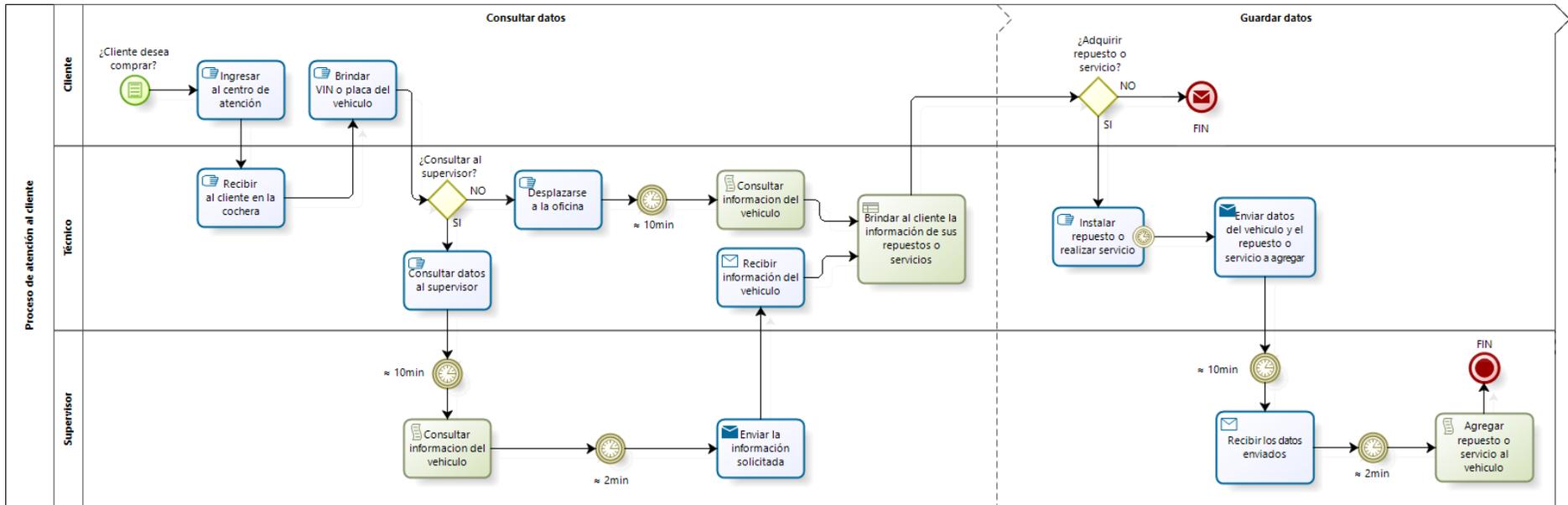


Figura 1. Proceso de atención al cliente AsIs

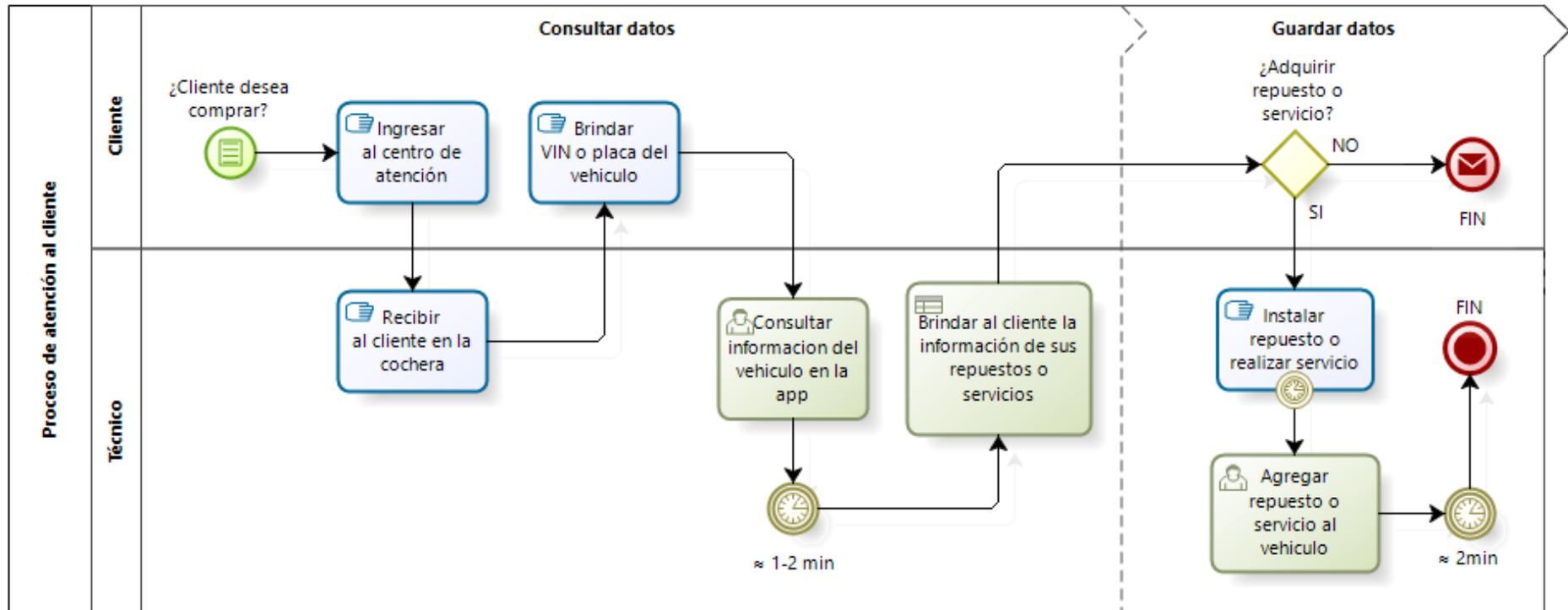


Figura 2. Proceso de atención al cliente ToBe

En el proceso inicial (AsIs). El cliente ingresa al centro de atención. Tras ser recibido por un técnico, el cliente le brinda su VIN (Vehicle Identification Number) o placa del vehículo. En este punto, el técnico puede elegir si consultar o no al supervisor.

- Si elige sí, deberá esperar su disponibilidad debido a que el supervisor tiene varios técnicos a cargo, esto suele tardar aproximadamente 10 minutos. Recolectar la información solicitada y enviarla al técnico que la solicitó tarda aproximadamente 2 min. Finalmente, el técnico recibe la información.

- Si elige no, deberá desplazarse hacia la oficina, lo cual toma aproximadamente 10 minutos debido a que la cochera es muy grande porque entran camiones y tarda en desplazarse. Finalmente, el técnico consulta personalmente los datos del vehículo.

A continuación, el técnico brinda al cliente la información sobre su vehículo, su garantía, los repuestos y/o servicios comprados. El cliente puede elegir si va a adquirir o no un nuevo servicio o repuesto para su vehículo.

- Si elige sí, el técnico debe instalar el repuesto solicitado o realizar el servicio. Tras terminar en x minutos, le envía los datos del vehículo y del servicio o repuesto que adquirió, al supervisor. Tras esperar su disponibilidad, aproximadamente 10 minutos el supervisor recibe los datos. Finalmente, los registra, lo cual tarda aproximadamente 2 minutos.

- Si elige no, el proceso termina.

En el proceso mejorado (ToBe). El cliente ingresa al centro de atención. Tras ser recibido por un técnico, el cliente le brinda su VIN o placa del vehículo. A diferencia del AsIs el técnico solicita personalmente los datos del vehículo por medio del aplicativo móvil, lo cual ahorra el tiempo de desplazamiento o el tiempo de espera de disponibilidad. Consultar la información por medio del aplicativo móvil solo tarda 1 o 2 minutos aproximadamente. A continuación, el técnico brinda al cliente la información sobre su vehículo, su garantía, los repuestos y/o servicios comprados. El cliente puede elegir si va a adquirir o no un nuevo servicio o repuesto para su vehículo.

- Si elige sí, el técnico debe instalar el repuesto solicitado o realizar el servicio. Tras terminar en x minutos, el técnico registra el servicio o repuesto adquirido, lo cual tarda aproximadamente 2 minutos.
- Si elige no, el proceso termina.

Analizar requerimientos del aplicativo móvil

Para el desarrollo del aplicativo se tuvieron que identificar ciertos requerimientos previos al desarrollo del aplicativo móvil, los cuales servirán como guía o referencia para que el aplicativo cumpla con las funciones necesarias.

Tabla 2.

Requerimientos identificados para el desarrollo del aplicativo móvil

N°	Caso de uso	Requerimiento	Prototipo	Versión
1	Registrarse	Registrarse para poder iniciar sesión en el aplicativo	NO	v1.0
2	Iniciar sesión	Iniciar sesión por medio de un usuario registrado	NO	v1.0
3	Iniciar sesión con Google	Iniciar sesión por medio de una cuenta de Google	NO	v1.0
4	Consultar datos del vehículo	Consultar los datos del vehículo por placa	SI	v1.0
5	Consultar repuestos	Consultar repuestos del vehículo de la placa ingresada	SI	v1.0
6	Consultar garantía	Consultar garantía de cada uno de los repuestos	SI	V2.0
7	Consultar servicios	Consultar servicios del vehículo de la placa ingresada	SI	V2.0
8	Agregar repuesto	Agregar repuesto al vehículo de placa ingresada	SI	v2.0
9	Agregar servicio	Agregar servicio al vehículo de placa ingresada	SI	v2.0
10	Generar QR	Generar un nuevo QR de la placa de los clientes	NO	v3.0
11	Compartir QR	Compartir el QR generado a una impresora o al WhatsApp del cliente dueño de la placa	NO	v3.0
12	Escanear QR	Escanear un QR en lugar de ingresar la placa manualmente	NO	v3.0

Fuente: Elaboración propia

Preparar prototipos previos al desarrollo del aplicativo móvil

Previo al desarrollo del aplicativo móvil se diseñaron algunos prototipos en los cuales se puede ver el diseño que tendrían las interfaces.



Figura 3. Prototipo de Iniciar sesión. En esta pantalla el colaborador puede registrarse o iniciar sesión, por medio de un correo o mediante una sesión iniciada con una cuenta de Google.

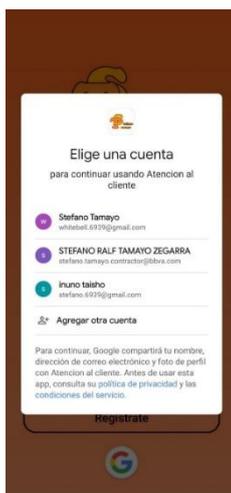


Figura 4. Prototipo de Iniciar sesión con Google. Desde aquí el colaborador puede seleccionar alguna de las cuentas de Google que tenga o agregar una nueva para acceder al aplicativo.



Figura 5. Prototipo de Registrarse. Desde esta pantalla el colaborador podrá realizar el registro de su cuenta mediante un correo.



Figura 6. Prototipo de Inicio. Esta es la pantalla de inicio, donde se muestra una bienvenida y las distintas opciones que tiene el aplicativo. Además del logo de la empresa y el botón para cerrar sesión.



Figura 7. Prototipo de Consultar datos del vehículo. En esta pantalla se muestran los datos del vehículo, cuya placa haya sido ingresada para la consulta. Además, muestra 2 botones para poder ingresar al detalle de los repuestos o vehículos pertenecientes al vehículo en cuestión.



Figura 8. Prototipo de Consultar repuestos. En esta pantalla se muestran los repuestos instalados a un vehículo en particular. Permite la posibilidad de buscar repuestos. Indica los datos de los repuestos, como nombre, marca, fecha, técnico, tiempo de garantía y si la garantía del repuesto se encuentra vigente o no. También muestra un botón para agregar un nuevo repuesto.



Figura 9. Prototipo de Agregar repuesto. En esta pantalla se pueden agregar repuestos, seleccionando la categoría y posteriormente el repuesto. Por cada repuesto seleccionado, mostrará la descripción de este y un botón para agregar este repuesto, de ser el caso.



Figura 10. Prototipo de Consultar Servicios En esta pantalla se muestran los servicios aplicados a un vehículo en particular. Permite la posibilidad de buscar servicios. Indica los datos de los servicios, como nombre, tipo, fecha, técnico y sede. También muestra un botón para agregar un nuevo servicio.



Figura 11. Prototipo de Agregar servicios. En esta pantalla se pueden agregar servicios, seleccionando la categoría y posteriormente el servicio. Por cada servicio seleccionado, mostrará la descripción de este y un botón para agregar este servicio, de ser el caso.



Figura 12. Prototipo de Generar QR. En esta pantalla se puede genera un código QR para la placa del vehículo de un cliente. Además, tiene un botón para poder compartir el código QR generado con el cliente, para facilitar próximas atenciones.

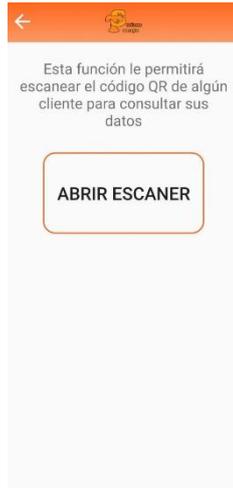


Figura 13. Prototipo de Escanear QR. A partir de esta pantalla se podrá ejecutar el escáner, abriendo la cámara para aplicar el reconocimiento de imágenes en el QR del cliente.

Modelar la base de datos del aplicativo móvil

Se realizó el modelado físico y lógico de la base de datos para el aplicativo móvil, este modelo lógico corresponde a una base de datos no estructurada.

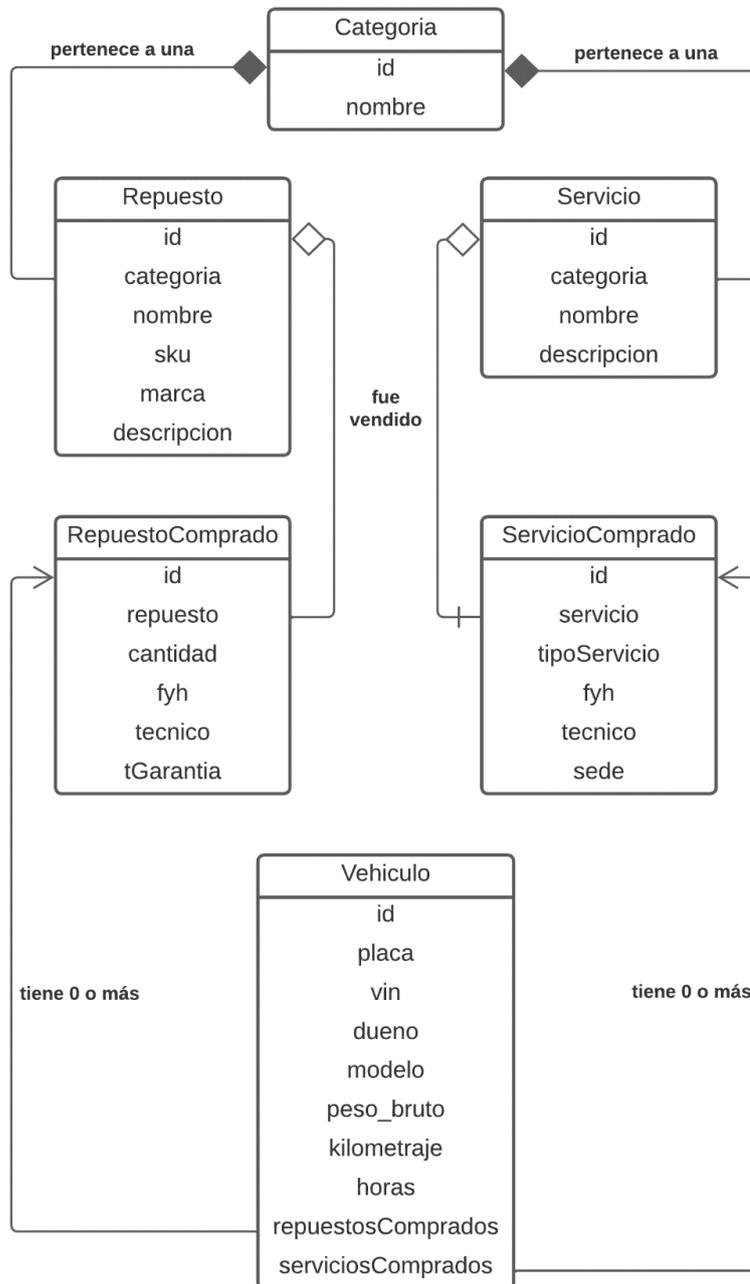


Figura 14. Modelo lógico de la base de datos

Ahora se muestra el modelo físico de base de datos no estructurada para el aplicativo móvil.

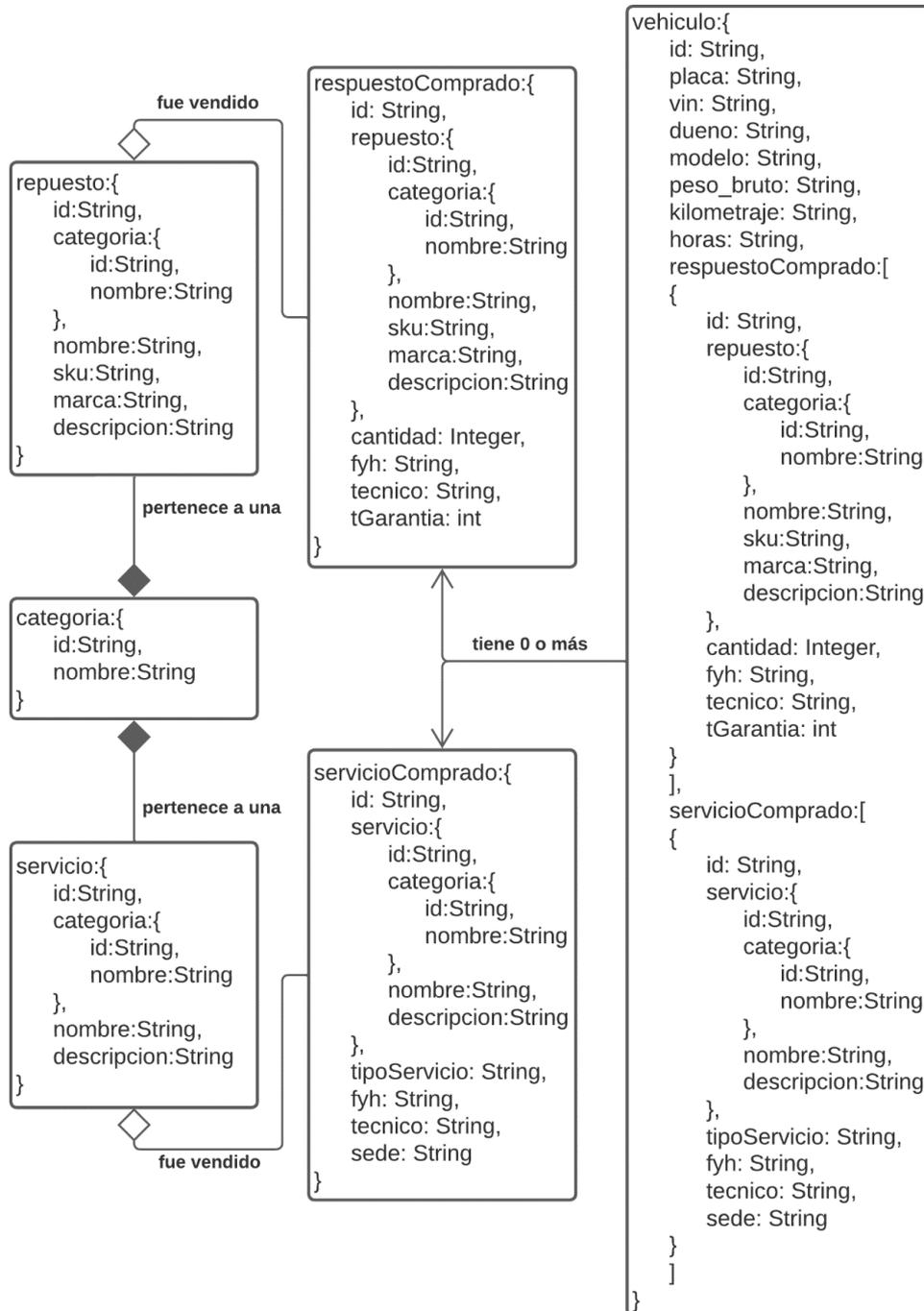
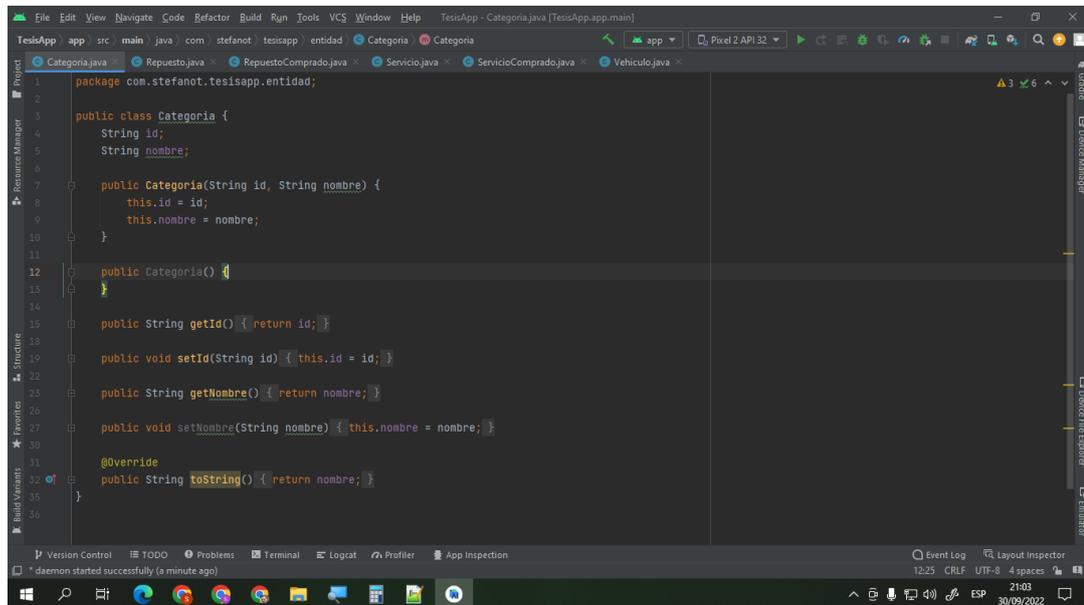


Figura 15. Modelo físico de la base de datos

Crear las tablas o entidades de la base de datos

Se crearon las entidades con sus relaciones respectivas para su uso como base datos no relacional. Se muestran capturas de las entidades creadas como clases.



```
package com.stefanot.tesisapp.entidad;

public class Categoria {
    String id;
    String nombre;

    public Categoria(String id, String nombre) {
        this.id = id;
        this.nombre = nombre;
    }

    public Categoria() {}

    public String getId() { return id; }

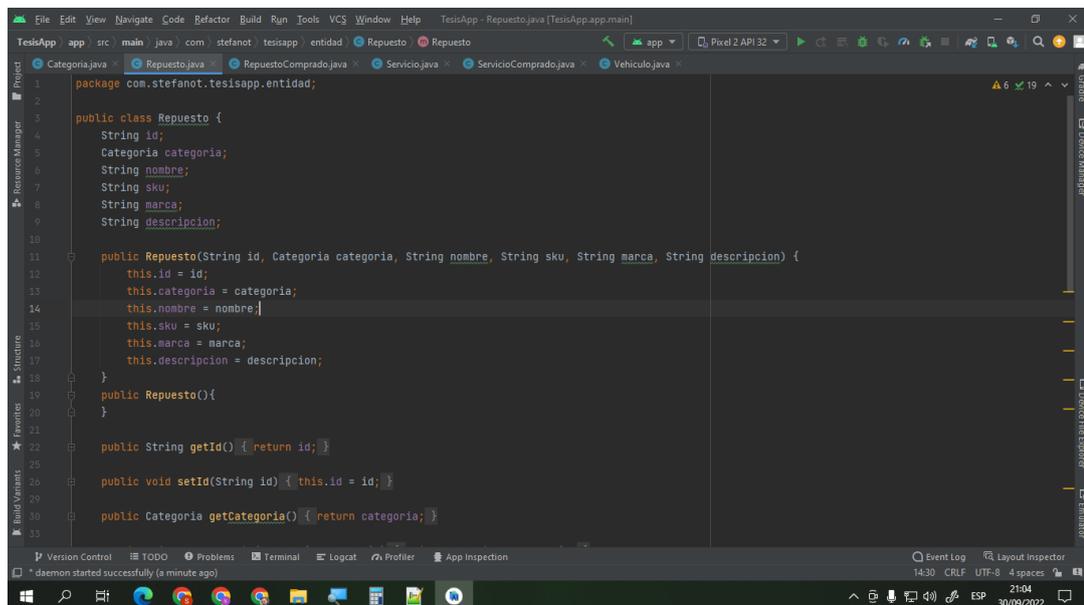
    public void setId(String id) { this.id = id; }

    public String getNombre() { return nombre; }

    public void setNombre(String nombre) { this.nombre = nombre; }

    @Override
    public String toString() { return nombre; }
}
```

Figura 16. Entidad Categoría



```
package com.stefanot.tesisapp.entidad;

public class Repuesto {
    String id;
    Categoria categoria;
    String nombre;
    String sku;
    String marca;
    String descripcion;

    public Repuesto(String id, Categoria categoria, String nombre, String sku, String marca, String descripcion) {
        this.id = id;
        this.categoria = categoria;
        this.nombre = nombre;
        this.sku = sku;
        this.marca = marca;
        this.descripcion = descripcion;
    }

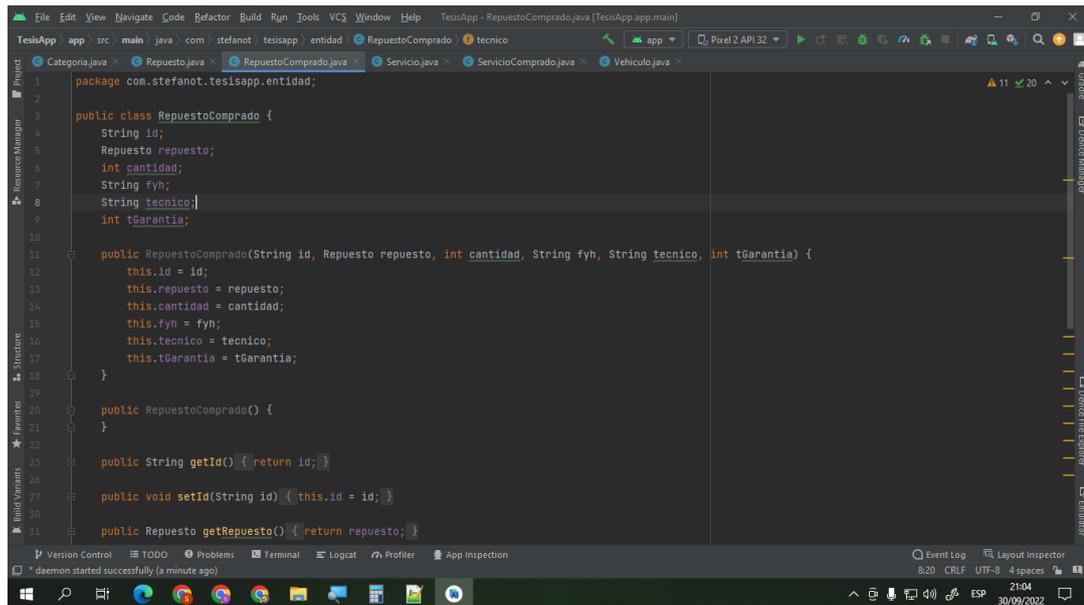
    public Repuesto() {}

    public String getId() { return id; }

    public void setId(String id) { this.id = id; }

    public Categoria getCategoria() { return categoria; }
}
```

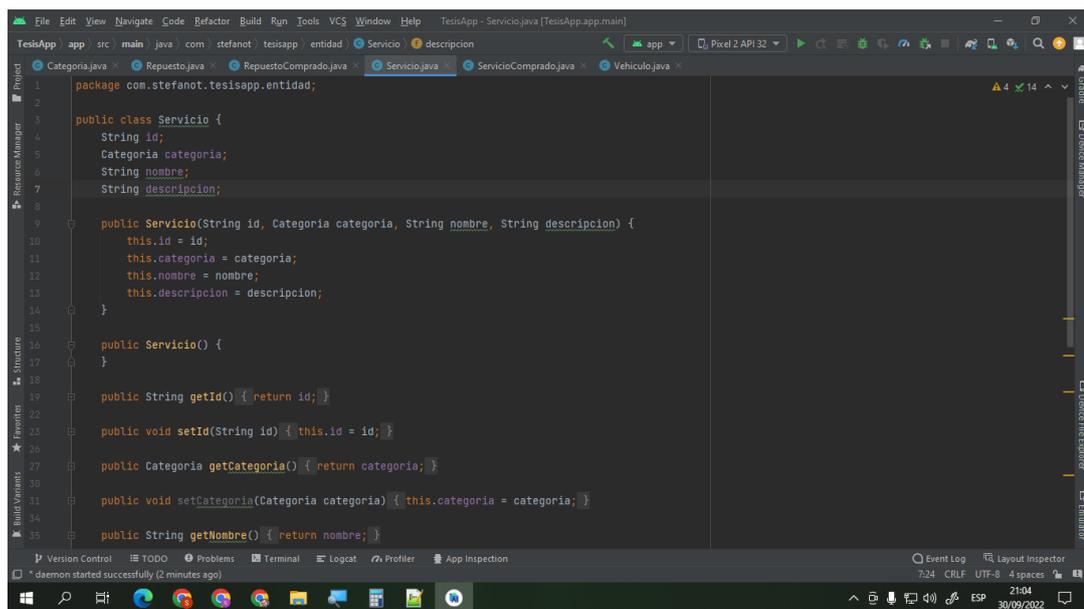
Figura 17. Entidad Repuesto



```

1 package com.stefanot.tesisapp.entidad;
2
3 public class RepuestoComprado {
4     String id;
5     Repuesto repuesto;
6     int cantidad;
7     String fyh;
8     String tecnico;
9     int tGarantia;
10
11     public RepuestoComprado(String id, Repuesto repuesto, int cantidad, String fyh, String tecnico, int tGarantia) {
12         this.id = id;
13         this.repuesto = repuesto;
14         this.cantidad = cantidad;
15         this.fyh = fyh;
16         this.tecnico = tecnico;
17         this.tGarantia = tGarantia;
18     }
19
20     public RepuestoComprado() {
21     }
22
23     public String getId() { return id; }
24
25     public void setId(String id) { this.id = id; }
26
27     public Repuesto getRepuesto() { return repuesto; }
28
29 }
    
```

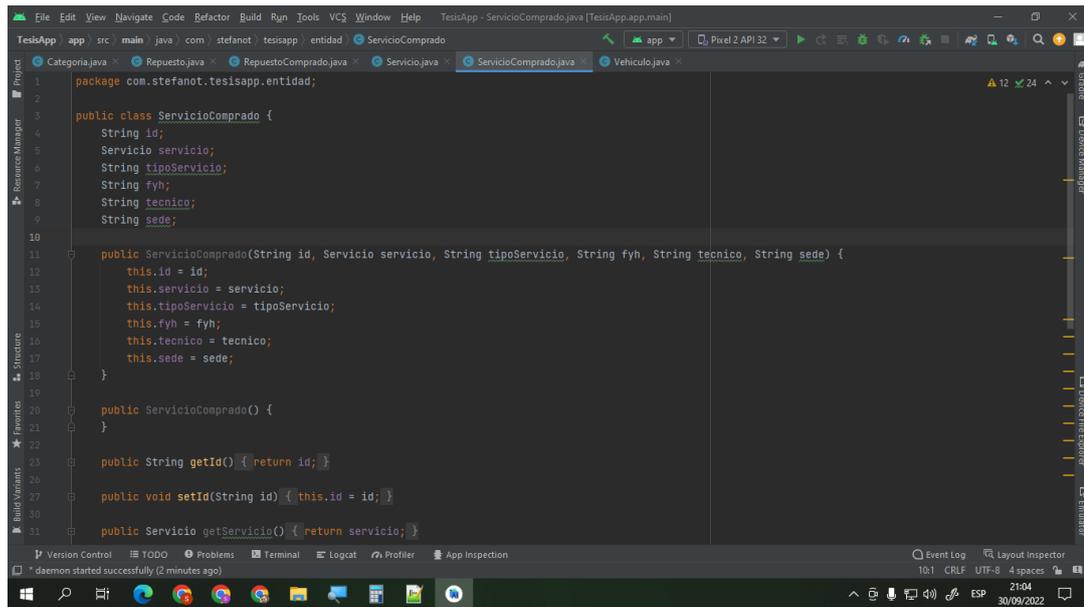
Figura 18. Entidad Repuesto Comprado



```

1 package com.stefanot.tesisapp.entidad;
2
3 public class Servicio {
4     String id;
5     Categoria categoria;
6     String nombre;
7     String descripcion;
8
9     public Servicio(String id, Categoria categoria, String nombre, String descripcion) {
10         this.id = id;
11         this.categoria = categoria;
12         this.nombre = nombre;
13         this.descripcion = descripcion;
14     }
15
16     public Servicio() {
17     }
18
19     public String getId() { return id; }
20
21     public void setId(String id) { this.id = id; }
22
23     public Categoria getCategoria() { return categoria; }
24
25     public void setCategoria(Categoria categoria) { this.categoria = categoria; }
26
27     public String getNombre() { return nombre; }
28
29 }
    
```

Figura 19. Entidad Servicio



```

package com.stefanot.tesisapp.entidad;

public class ServicioComprado {
    String id;
    Servicio servicio;
    String tipoServicio;
    String fyh;
    String tecnico;
    String sede;

    public ServicioComprado(String id, Servicio servicio, String tipoServicio, String fyh, String tecnico, String sede) {
        this.id = id;
        this.servicio = servicio;
        this.tipoServicio = tipoServicio;
        this.fyh = fyh;
        this.tecnico = tecnico;
        this.sede = sede;
    }

    public ServicioComprado() {
    }

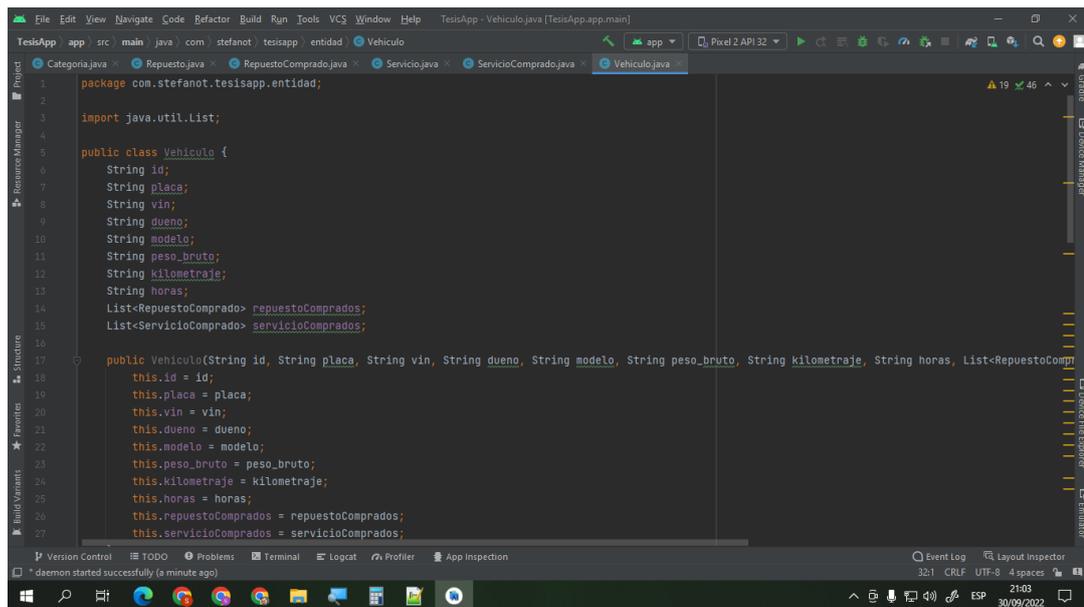
    public String getId() { return id; }

    public void setId(String id) { this.id = id; }

    public Servicio getServicio() { return servicio; }
}

```

Figura 20. Entidad Servicio Comprado



```

package com.stefanot.tesisapp.entidad;

import java.util.List;

public class Vehiculo {
    String id;
    String placa;
    String vin;
    String dueno;
    String modelo;
    String peso_bruto;
    String kilometraje;
    String horas;
    List<RepuestoComprado> repuestoComprados;
    List<ServicioComprado> servicioComprados;

    public Vehiculo(String id, String placa, String vin, String dueno, String modelo, String peso_bruto, String kilometraje, String horas, List<RepuestoComprado> repuestoComprados, List<ServicioComprado> servicioComprados) {
        this.id = id;
        this.placa = placa;
        this.vin = vin;
        this.dueno = dueno;
        this.modelo = modelo;
        this.peso_bruto = peso_bruto;
        this.kilometraje = kilometraje;
        this.horas = horas;
        this.repuestoComprados = repuestoComprados;
        this.servicioComprados = servicioComprados;
    }
}

```

Figura 21. Entidad Vehículo

Desarrollar el front-end del aplicativo móvil

Se muestran el front-end del aplicativo móvil desarrollado para cumplir con el objetivo de la investigación. El front-end es la parte con la que interactúa el usuario al usar

el aplicativo.

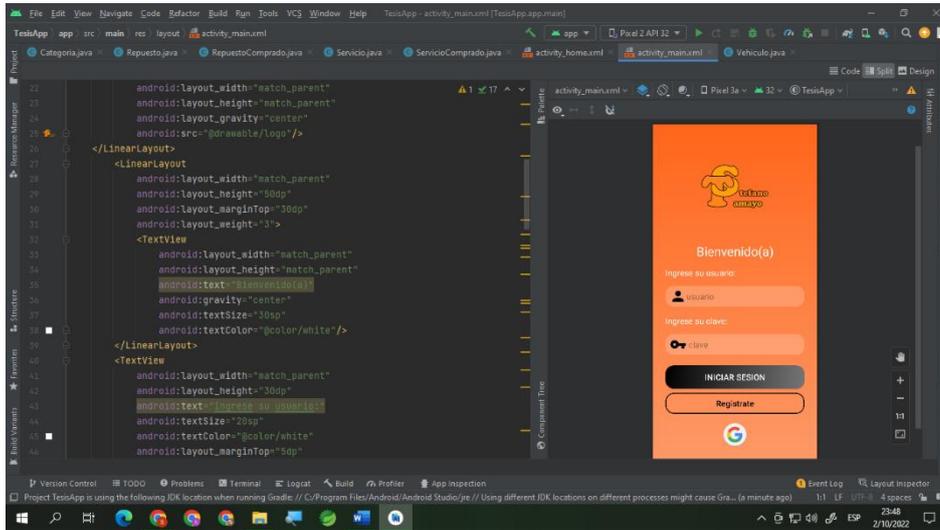


Figura 22. Captura de front-end de Inicio de sesión

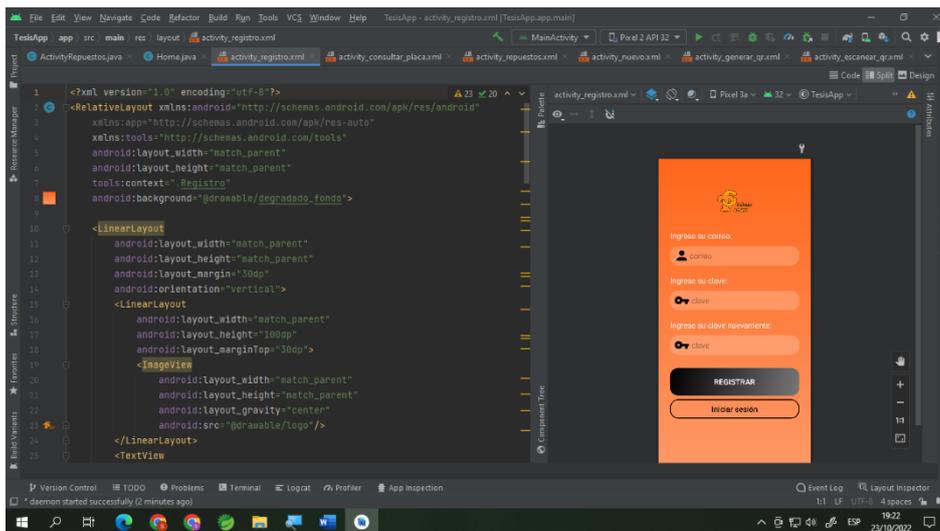


Figura 23. Captura de front-end de Registro

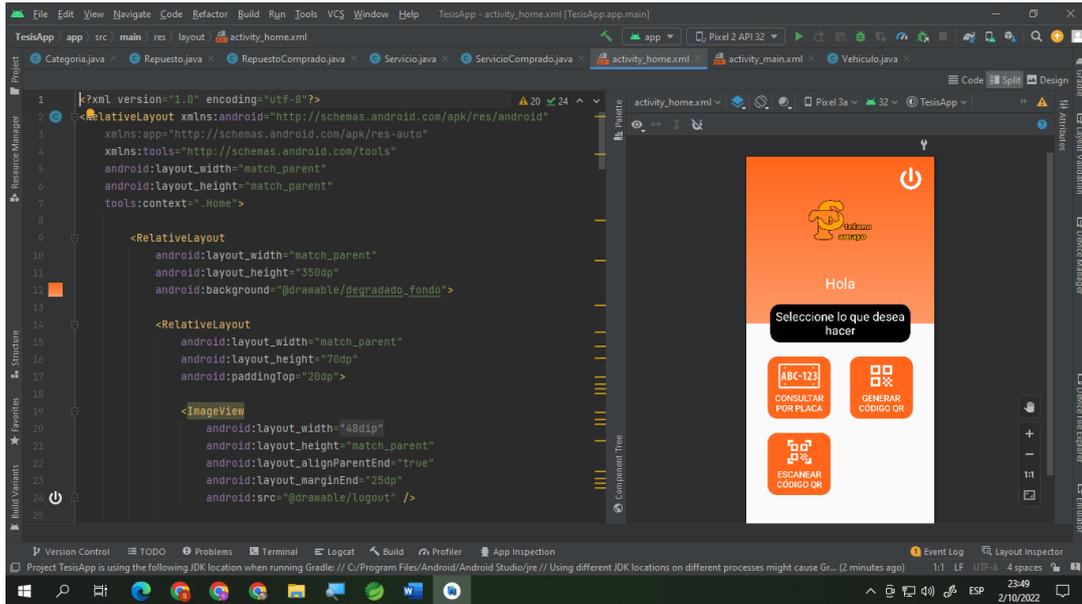


Figura 24. Captura de front-end de Home

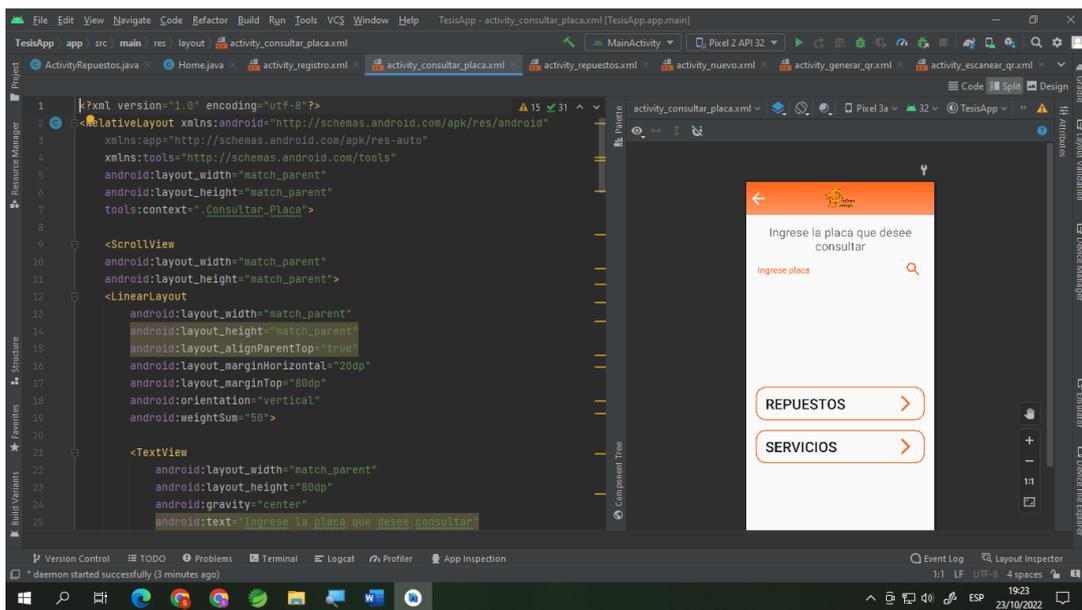


Figura 25. Captura de front-end de Consultar placa

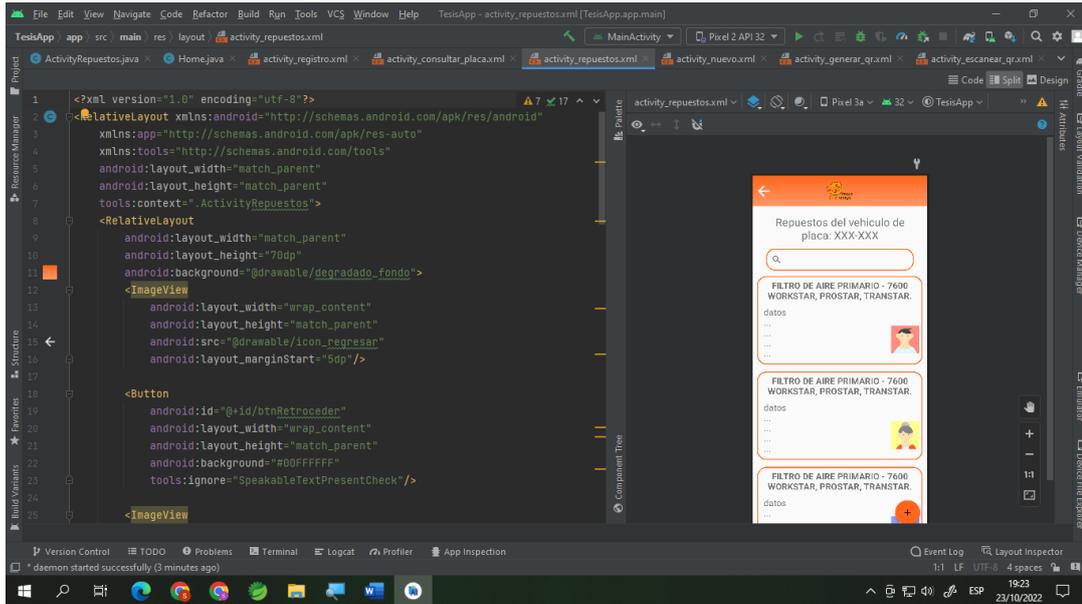


Figura 26. Captura de front-end de Consultar repuestos o servicios

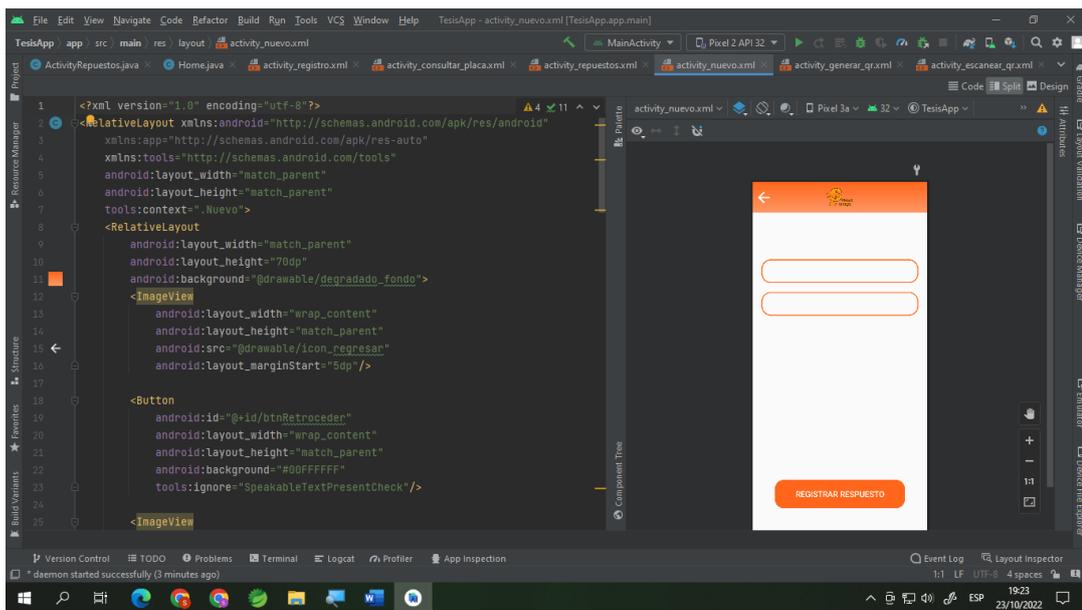


Figura 27. Captura de front-end de Agregar repuesto o servicio

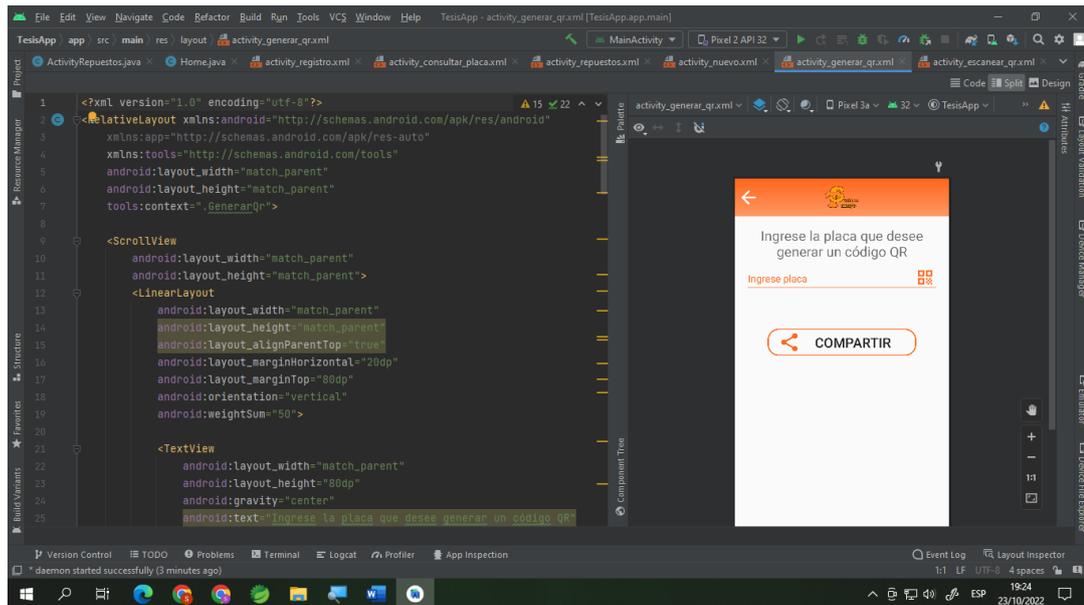


Figura 28. Captura de front-end de Generar QR

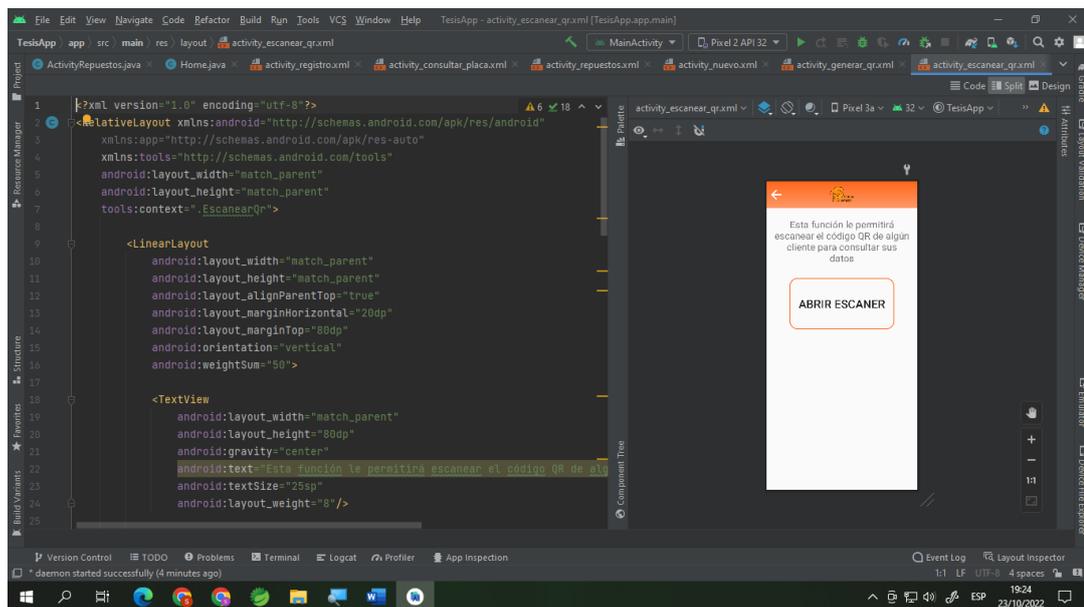


Figura 29. Captura de front-end de Escanear QR

Realizar las pruebas, para validar las funcionalidades del aplicativo móvil

Para validar el aplicativo móvil desarrollado se usaron diferentes equipos móviles de distintas gamas (alta, media y baja), se probaron todas las funcionalidades con éxito.

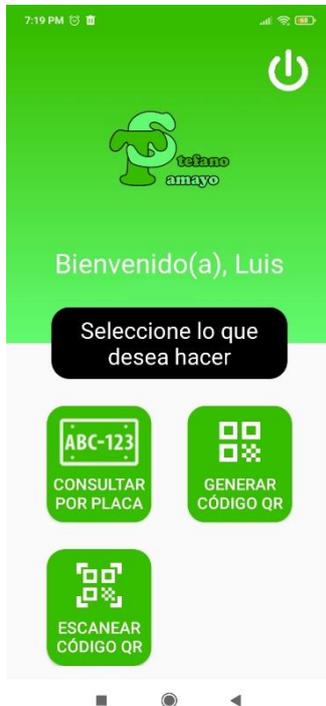


Figura 30. Prueba en dispositivo físico de Home



Figura 31. Prueba en dispositivo físico de Consultar placa



Figura 32. Prueba en dispositivo físico de Consultar repuestos



Figura 33. Prueba en dispositivo físico de Agregar repuesto



Figura 34. Prueba en dispositivo físico de Consultar servicio

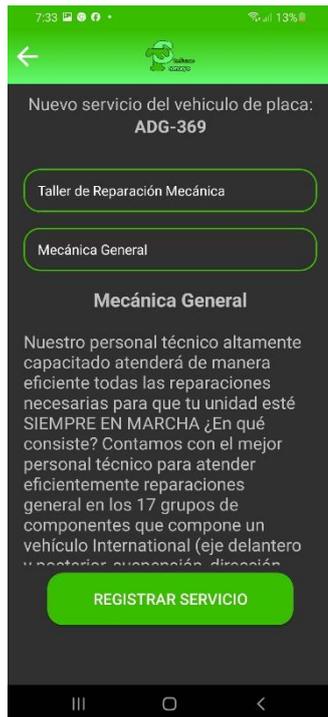


Figura 35. Prueba en dispositivo físico de Agregar servicio



Figura 36. Prueba en dispositivo físico de Generar QR

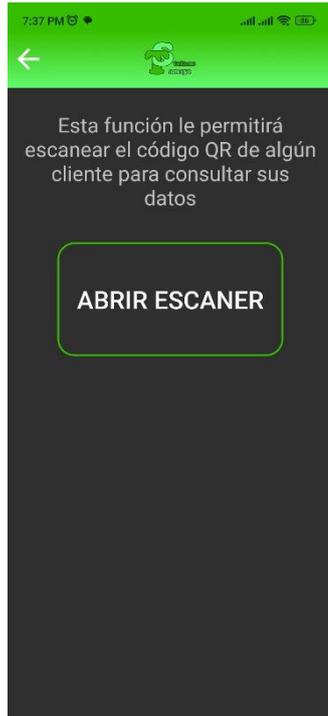


Figura 37. Prueba en dispositivo físico de Escanear QR



Figura 38. Prueba en dispositivo físico de Escáner funcionando

Además, se realizaron pruebas en dispositivos Android virtuales con ayuda de Firebase, el cual nos provee de un *Test Lab*. Igualmente se usaron dispositivos de gama alta, media y baja para comprobar la funcionalidad del aplicativo móvil desarrollado.

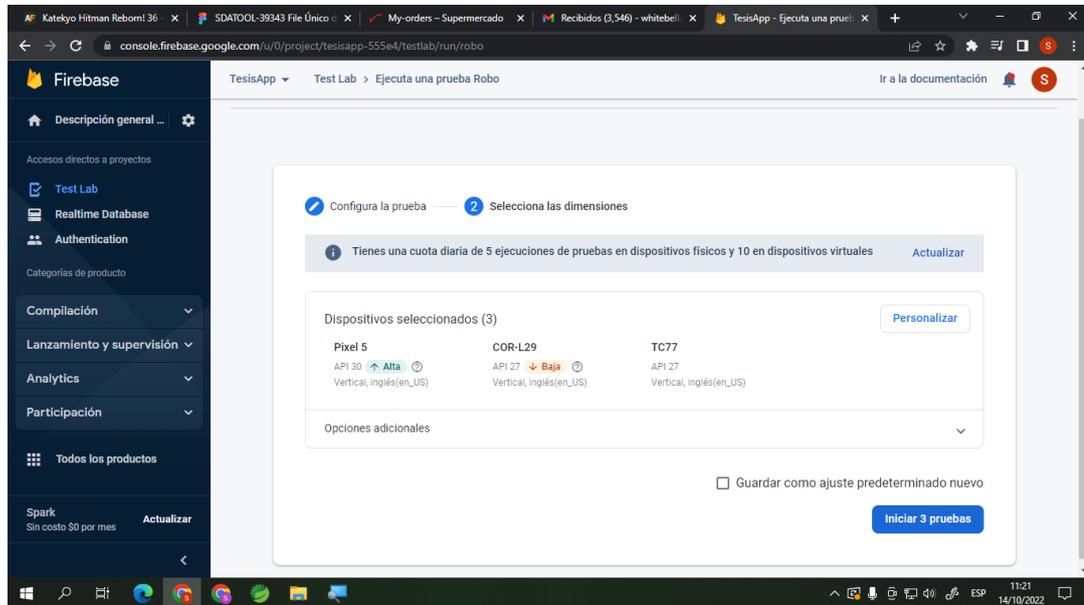


Figura 39. Captura de selección de dispositivos a probar

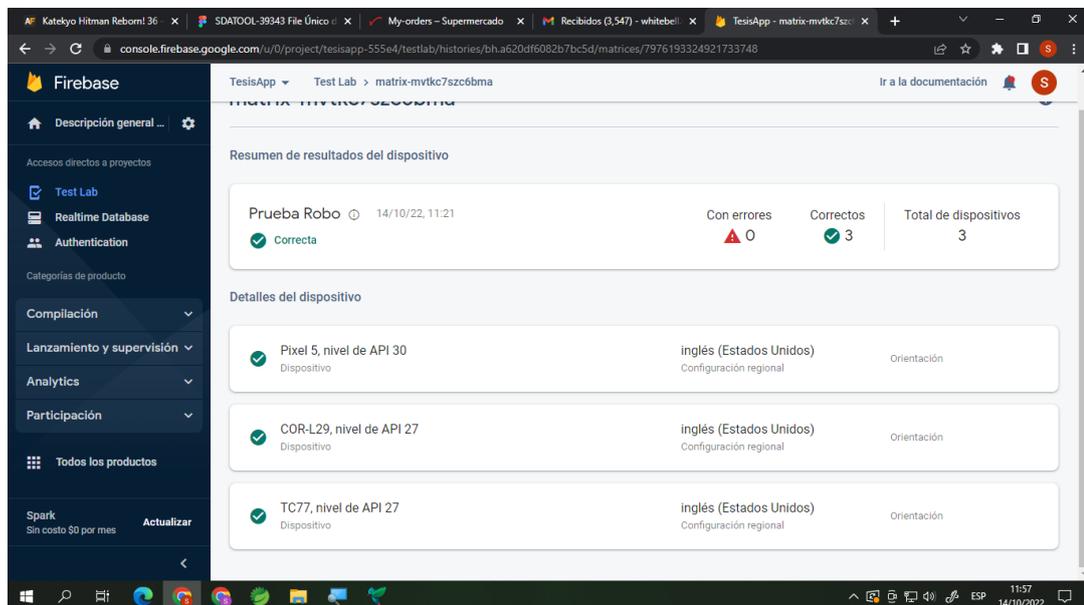


Figura 40. Resultados de las pruebas en Firebase

Presentar el aplicativo móvil a los usuarios

Se presentó el aplicativo móvil a los usuarios que forman parte de la población, se les explicó la funcionalidad y envió el archivo apk para que ellos puedan instalar la aplicación móvil en sus dispositivos. A continuación, el enlace que se envió a los usuarios para descargar el aplicativo móvil:

<https://drive.google.com/drive/folders/1hTDy2Z7hR5IGr3jdAUQqe15Z46RuSqF9?usp=sharing>

Recolectar opiniones sobre el uso del aplicativo móvil

Para recolectar las opiniones sobre el aplicativo móvil, se usó un formulario digital creado en Google Forms como se estableció en la definición del instrumento, se les envió el formulario por medios digitales. A continuación, el enlace del PDF del formulario hecho en Google Forms:

<https://drive.google.com/drive/folders/1NBWtkdNcYK-fSWpM7SUB9XA-J4Ec42kA?usp=sharing>

Hoja de datos de las respuestas del pre-test:

https://docs.google.com/spreadsheets/d/1IBmQ0sOkZEy479SdgZHXg4KFWIEQgUJhtHfU_Y7pUJg/edit?usp=share_link

Hoja de datos de las respuestas del post-test:

https://docs.google.com/spreadsheets/d/1J2fg3JXyPgbYdaPgcIvfnLQN0zZbKWg4Z0-GVZ_5oI0/edit?usp=share_link

3.2. Hallazgos

Se realizó el análisis de las respuestas antes del desarrollo del aplicativo móvil (pre-test) y después del desarrollo del aplicativo móvil (post-test). Estas respuestas fueron obtenidas de la encuesta realizada a los colaboradores de una empresa del rubro automotriz de la ciudad de Lima, 2022. A continuación, se muestran los resultados obtenidos por cada pregunta:

Pregunta 1: ¿Requiere desplazarse a un centro de datos o consultar con otra persona para obtener la información de un vehículo?

Tabla 3.

Análisis de frecuencia de la Pregunta 1 (accesibilidad a los datos) (pre-test)

		Frecuencia	Porcentaje
Válido	SI	8	100.0

Fuente: Elaboración propia

Tabla 4.

Análisis de frecuencia de la Pregunta 1 (accesibilidad a los datos) (post-test)

		Frecuencia	Porcentaje
Válido	SI	7	87.5
	NO	1	12.5
	Total	8	100.0

Fuente: Elaboración propia

Se puede observar que antes del desarrollo del aplicativo, todos los colaboradores (100.0%) necesitaban desplazarse o consultar para obtener la información de un vehículo. Después, del aplicativo observamos que al menos una persona (12.5%), ya no requiere hacerlo, mientras que los demás (87.5%) aún requieren de realizar alguna consulta. Estos

resultados no son del todo satisfactorios, pero se comprende que aún exista dicha resistencia al cambio al implementar una nueva solución. También, se contempla la posibilidad de que los encuestados no hayan leído adecuadamente la pregunta.

3.2.1. Hallazgos de la primera hipótesis específica

Este hallazgo respondió a la primera hipótesis específica de la investigación, la cual fue: La aplicación móvil mejora el tiempo de obtención de datos de un vehículo, en una empresa automotriz.

Pregunta 2: ¿Cuánto tiempo tarda en promedio en obtener los datos de un vehículo, desde que llega? (Incluyendo tiempo de desplazamiento a la oficina o el tiempo de respuesta de la persona que consultó)

Tabla 5.

Análisis de frecuencia de la Pregunta 2 (datos del vehículo) (pre-test)

		Frecuencia	Porcentaje
Válido	Aprox. 8 minutos	1	12.5
	Aprox. 9 minutos	1	12.5
	Aprox. 10 minutos	2	25.0
	Aprox. 11 minutos	1	12.5
	Aprox. 12 minutos	3	37.5
	Total	8	100.0

Fuente: Elaboración propia

Tabla 6.

Análisis de frecuencia de la Pregunta 2 (datos del vehículo) (post-test)

	Frecuencia	Porcentaje
Válido Aprox. 3-4 minutos	3	37.5
Aprox. 5-6 minutos	3	37.5
Aprox. 12 minutos	2	25.0
Total	8	100.0

Fuente: Elaboración propia

En el pre-test se obtuvieron los siguientes resultados, con respecto al tiempo de obtención de los datos de un vehículo: 12.5% dice que tarda 8 minutos, otro 12.5% dice que tarda 9 minutos, el 25% de los encuestados dice que tarda 10 minutos. Otro 12.5% dice que tarda 11 minutos y un 37.5% dice que tarda aproximadamente 12 minutos. Con respecto al post-test, los resultados son: el 37.5% dice que tarda entre 3 y 4 minutos, otro 37.5% dice que tarda entre 5 y 6 minutos, y el 25% sostiene que tarda aproximadamente 12 minutos.

Tras realizar la implementación del aplicativo, se observa una reducción en los tiempos de obtención de los datos del vehículo que está siendo atendido. Es decir, que la hipótesis fue validada.

3.2.2. Hallazgos de la segunda hipótesis específica

Este hallazgo respondió a la segunda hipótesis específica de la investigación, la cual fue: La aplicación móvil mejora el tiempo de obtención de datos de los servicios aplicados, en una empresa automotriz.

Pregunta 3: ¿Cuánto tiempo tarda en promedio en obtener los datos de servicios hechos a un vehículo, desde que llega? (Incluyendo tiempo de desplazamiento a la oficina o el tiempo de respuesta de la persona que consultó)

Tabla 7.

Análisis de frecuencia de la Pregunta 3 (datos de servicios) (pre-test)

		Frecuencia	Porcentaje
Válido	Aprox. 9 minutos	2	25.0
	Aprox. 10 minutos	3	37.5
	Aprox. 11 minutos	1	12.5
	Aprox. 12 minutos	2	25.0
	Total	8	100.0

Fuente: Elaboración propia

Tabla 8.

Análisis de frecuencia de la Pregunta 3 (datos de servicios) (post-test)

		Frecuencia	Porcentaje
Válido	Aprox. 3-4 minutos	2	25.0
	Aprox. 5-6 minutos	4	50.0
	Aprox. 12 minutos	2	25.0
	Total	8	100.0

Fuente: Elaboración propia

En el pre-test se obtuvieron los siguientes resultados, con respecto al tiempo de obtención de los datos de los servicios de un vehículo: 25% dice que tarda 9 minutos, el 37.5% de los encuestados dice que tarda 10 minutos, un 12.5% dice que tarda 11 minutos. Finalmente, otro 25% dice que tarda aproximadamente 12 minutos. Con respecto al post-test, los resultados son: el 25% dice que tarda entre 3 y 4 minutos, el 50% dice que tarda entre 5 y 6 minutos, y el 25% sostiene que tarda aproximadamente 12 minutos.

Tras realizar la implementación del aplicativo, se observa una reducción en los tiempos de obtención de datos de los servicios que han sido realizados previamente al vehículo del cliente. Lo que significa que la hipótesis fue validada.

3.2.3. Hallazgos de la tercera hipótesis específica

Este hallazgo respondió a la tercera hipótesis específica de la investigación, la cual fue: La aplicación móvil mejora el tiempo de obtención de datos de los repuestos instalados, en una empresa automotriz.

Pregunta 4: ¿Cuánto tiempo tarda en promedio en obtener los datos de repuestos instalados a un vehículo, desde que llega? (Incluyendo tiempo de desplazamiento a la oficina o el tiempo de respuesta de la persona que consultó)

Tabla 9.

Análisis de frecuencia de la Pregunta 4 (datos de repuestos) (pre-test)

		Frecuencia	Porcentaje
Válido	Aprox. 9 minutos	2	25.0
	Aprox. 10 minutos	2	25.0
	Aprox. 11 minutos	1	12.5
	Aprox. 12 minutos	3	37.5
	Total	8	100.0

Fuente: Elaboración propia

Tabla 10.

Análisis de frecuencia de la Pregunta 4 (datos de repuestos) (post-test)

	Frecuencia	Porcentaje
Válido Aprox. 1-2 minutos	1	12.5
Aprox. 3-4 minutos	2	25.0
Aprox. 5-6 minutos	3	37.5
Aprox. 12 minutos	2	25.0
Total	8	100.0

Fuente: Elaboración propia

En el pre-test se obtuvieron los siguientes resultados, con respecto al tiempo de obtención de los datos de los servicios de un vehículo: 25% dice que tarda 9 minutos, otro 25% dice que tarda 10 minutos, un 12.5% dice que tarda 11 minutos. Finalmente, el 37.5% de los encuestados dice que tarda aproximadamente 12 minutos. Con respecto al post-test, los resultados son: el 12.5% dice que tarda entre 1 y 2 minutos, el 25% dice que tarda entre 3 y 4 minutos, el 37.5% dice que tarda entre 5 y 6 minutos, y el 25% sostiene que tarda aproximadamente 12 minutos.

Tras realizar la implementación del aplicativo, se observa una reducción en los tiempos de obtención de datos de los repuestos que han sido instalados previamente al vehículo del cliente. Es decir, la hipótesis fue validada.

3.2.4. Hallazgos de la cuarta hipótesis específica

Este hallazgo respondió a la cuarta hipótesis específica de la investigación, la cual fue: La aplicación móvil mejora el tiempo de registro de datos de los repuestos y servicios,

en una empresa automotriz.

Pregunta 5: ¿Cuánto tiempo tarda en promedio en registrar un repuesto o servicio a un vehículo, desde que llega? (Incluyendo tiempo de desplazamiento a la oficina o el tiempo de respuesta de la persona que consultó)

Tabla 11.

Análisis de frecuencia de la Pregunta 5 (registrar) (pre-test)

		Frecuencia	Porcentaje
Válido	Aprox. 9 minutos	1	12.5
	Aprox. 10 minutos	3	37.5
	Aprox. 11 minutos	2	25.0
	Aprox. 12 minutos	2	25.0
	Total	8	100.0

Fuente: Elaboración propia

Tabla 12.

Análisis de frecuencia de la Pregunta 5 (registrar) (post-test)

		Frecuencia	Porcentaje
Válido	Aprox. 3-4 minutos	1	12.5
	Aprox. 5-6 minutos	5	62.5
	Aprox. 12 minutos	2	25.0
	Total	8	100.0

Fuente: Elaboración propia

En el pre-test se obtuvieron los siguientes resultados, con respecto al tiempo de guardado de los datos de los repuestos o servicios de un vehículo: 12.5% dice que tarda 9 minutos, el 37.5% dice que tarda 10 minutos, un 25% dice que tarda 11 minutos y otro 25% dice que tarda aproximadamente 12 minutos. Con respecto al post-test, los resultados son: el 12.5% dice que tarda entre 3 y 4 minutos, el 62.5% dice que tarda entre 5 y 6

minutos. Finalmente, el 25% sostiene que tarda aproximadamente 12 minutos.

Tras realizar la implementación del aplicativo, se observa una reducción en los tiempos de registro de datos de un nuevo repuesto o servicio que ha sido aplicado en la atención, para el vehículo del cliente. Cabe recalcar que, el registro tarda un poco más debido a los campos a completar. Se valida la hipótesis.

3.2.5. Hallazgos de la hipótesis general

Este hallazgo respondió a la hipótesis general de la investigación, la cual fue: La aplicación móvil mejora la atención al cliente en una empresa del rubro automotriz de la ciudad de Lima, 2022

Se realizó el cálculo del promedio de los tiempos por pregunta, con base a las respuestas obtenidas de los colaboradores. Solo se consideraron las preguntas que especifiquen el tiempo para realizar una actividad.

Tabla 13.

Promedio de tiempo por cada pregunta (pre-test)

Colaborador	Tiempo P2	Tiempo P3	Tiempo P4	Tiempo P5
1	10	10	12	10
2	12	12	12	10
3	12	12	12	12
4	11	11	10	11
5	10	10	11	11
6	12	9	9	12
7	9	9	9	10
8	8	10	10	9
Promedio	10.5	10.375	10.625	10.625

Fuente: Elaboración propia

Tabla 14.

Promedio de tiempo por cada pregunta (post-test)

Colaborador	Tiempo P2	Tiempo P3	Tiempo P4	Tiempo P5
1	5.5	5.5	5.5	5.5
2	3.5	3.5	3.5	5.5
3	5.5	3.5	5.5	3.5
4	5.5	5.5	5.5	5.5
5	12	12	12	12
6	3.5	5.5	3.5	5.5
7	12	12	12	12
8	3.5	5.5	1.5	5.5
Promedio	6.375	6.625	6.125	6.875

Fuente: Elaboración propia

Podemos observar que el tiempo promedio para consultar los datos de un vehículo (pregunta 2) sin el aplicativo es 10.5 minutos y con el aplicativo es 6.4 minutos. El tiempo promedio para consultar los servicios de un vehículo (pregunta 3) sin el aplicativo es 10.4 minutos y con el aplicativo es 6.6 minutos. El tiempo promedio para consultar los repuestos de un vehículo (pregunta 4) sin el aplicativo es 10.6 minutos y con el aplicativo es 6.1 minutos. Finalmente, el tiempo promedio para registrar los servicios o repuestos de un vehículo (pregunta 5) sin el aplicativo es 10.6 minutos y con el aplicativo es 6.9 minutos.

Adicionalmente, se hizo la comparación entre el promedio de los tiempos pre y post test de cada actividad especificada por pregunta y la reducción porcentual que refleja dicha comparación.

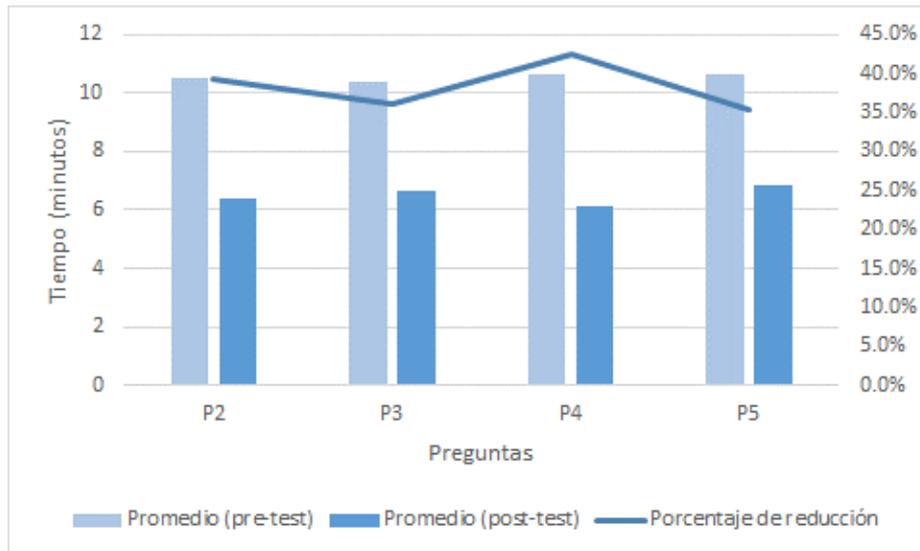


Figura 41. Comparación del promedio de tiempos entre pre-test y post-test

Asimismo, se realizó la comparación del promedio de los tiempos por respuestas de cada colaborador. Solo se consideraron las preguntas que especifiquen el tiempo para realizar una actividad.

Tabla 15.

Promedio de tiempo por cada colaborador (pre-test)

Colaborador	Tiempo P2	Tiempo P3	Tiempo P4	Tiempo P5	Promedio
1	10	10	12	10	10.5
2	12	12	12	10	11.5
3	12	12	12	12	12
4	11	11	10	11	10.75
5	10	10	11	11	10.5
6	12	9	9	12	10.5
7	9	9	9	10	9.25
8	8	10	10	9	9.25

Fuente: Elaboración propia

Tabla 16.

Promedio de tiempo por cada colaborador (post-test)

Colaborador	Tiempo P2	Tiempo P3	Tiempo P4	Tiempo P5	Promedio
1	5.5	5.5	5.5	5.5	5.5
2	3.5	3.5	3.5	5.5	4
3	5.5	3.5	5.5	3.5	4.5
4	5.5	5.5	5.5	5.5	5.5
5	12	12	12	12	12
6	3.5	5.5	3.5	5.5	4.5
7	12	12	12	12	12
8	3.5	5.5	1.5	5.5	4

Fuente: Elaboración propia

Podemos observar que, desde la percepción de los colaboradores encuestados, la mayoría representa una disminución en los tiempos. Sin embargo, hay algunos casos anómalos en el que este número se mantiene o incluso aumenta ligeramente.

3.2.6. Prueba de normalidad

Se realizó la prueba de normalidad para determinar qué tipo de prueba usar para probar la hipótesis con base en la significancia. En este caso, la población es menor a 30 y por ello se considera la significancia que otorga la prueba de Shapiro-Wilk.

Tabla 17.

Pruebas de normalidad

	Kolmogorov-Smirnova			Shapiro-Wilk		
	Estadístico	gl	Sig.	Estadístico	gl	Sig.
PreTest	.237	8	.200*	.910	8	.357
PostTest	.364	8	.002	.696	8	.002

Nota. * Esto es un límite inferior de la significación verdadera.

a. Corrección de significación de Lilliefors

Fuente: Elaboración propia

Como fue mencionado, se consideró la significancia de Shapiro-Wilk. Se observa que en el pre-test muestra una significancia de 0.357; sin embargo, en el post-test la significancia es de 0.002. Lo que significa que al ser inferior a 0.05, los datos no siguen una distribución normal.

3.2.7. Prueba no paramétrica

Debido a que los datos no siguen una distribución normal, para probar la hipótesis se tuvo que usar una prueba no paramétrica. La prueba que se usó en este caso fue la prueba de Wilcoxon. Para ello, se definió una hipótesis nula y la hipótesis planteada de la investigación.

H0: La aplicación móvil no mejora la atención al cliente en una empresa del rubro automotriz de la ciudad de Lima, 2022

Ha: La aplicación móvil mejora la atención al cliente en una empresa del rubro automotriz de la ciudad de Lima, 2022

Tabla 18.

Prueba de rangos con signo de Wilcoxon

	N	Rango promedio	Suma de rangos
Promedio de tiempo PostTest - Promedio de tiempo PreTest	6a	5.50	33.00
Rangos negativos	2b	1.50	3.00
Rangos positivos	0c		
Empates			
Total	8		

Nota.

a. Promedio de tiempo PostTest < Promedio de tiempo PreTest

- b. Promedio de tiempo PostTest > Promedio de tiempo PreTest
- c. Promedio de tiempo PostTest = Promedio de tiempo PreTest

Fuente: Elaboración propia

Tabla 19.

Prueba estadística de Wilcoxon

	Promedio de tiempo PostTest - Promedio de tiempo PreTest
Z	-2.106b
Sig. asin. (bilateral)	.035

Nota.

- b. Se basa en rangos positivos.

Fuente: Elaboración propia

En este caso, se rechaza la hipótesis nula porque la significancia de la prueba de Wilcoxon es de 0.035 y es menor que 0.05. Por lo tanto, aceptamos la hipótesis alternativa de que la aplicación móvil mejora la atención al cliente en una empresa del rubro automotriz de la ciudad de Lima, 2022.

3.3. Procedimientos tecnológicos

Con respecto a los objetivos específicos, estos fueron los procedimientos tecnológicos que se tomaron en cuenta para realizar todos los pasos para el cumplimiento de los objetivos planteados en la investigación.

Android Studio fue la herramienta principal que se usó para el desarrollo del aplicativo móvil planteado como solución de esta investigación. Este entorno de desarrollo suplió la necesidad de desarrollo tanto de front-end y back-end, proveyendo una amplia variedad de herramientas y fácil conexión con distintas librerías o servicios como los que

ofrece Firebase.

Firestore es una plataforma desarrollada por Google que provee diversas herramientas para el desarrollo móvil. Se usó debido a la necesidad de tener una base de datos en la nube cuya sincronización sea rápida y sus costos flexibles. Esta plataforma nos dio la posibilidad de almacenar los datos de la aplicación en una base de datos no relacional.

Emulador Bluestacks es un emulador de Android para pc el cual se usó para probar el aplicativo desarrollado. A pesar de que Android Studio cuenta con su propio emulador, se optó por el uso de este emulador para probar el aplicativo exportado como apk y en distintos entornos.

Tabla 20.

Recursos tecnológicos necesarios para la investigación

N°	Nombre	Aplicación
1	WINDOWS	Sistema operativo en el cual se instalaron los programas requeridos para el desarrollo
2	ANDROID STUDIO	Entorno de desarrollo usado para desarrollo y pruebas
3	FIREBASE	Plataforma que se usó para almacenar los datos en tiempo real
4	BLUESTACKS	Emulador que se usó para realizar las pruebas del aplicativo exportado

Fuente: Elaboración propia

JAVA fue el lenguaje de programación que se estableció para el desarrollo del aplicativo móvil en la parte del back-end. Se optó por el uso de este lenguaje debido a que se contaba con mayor experiencia en el uso de JAVA. Además, de la amplia documentación que se encuentra disponible sobre este lenguaje de programación.

HTML y **XML** fueron 2 lenguajes de marcado que se usaron principalmente para el desarrollo del front-end del aplicativo móvil. Se usó **XML** para el desarrollo de la totalidad de las interfaces desarrolladas para el aplicativo. Por otro lado, se usó **HTML** para la personalización de algunos textos dentro de las interfaces.

Librería Zxing fue la librería que se usó para apoyar al aplicativo en cuanto al reconocimiento y creación de códigos QR los cuales son generados para cada cliente y escaneados para agilizar aún más el proceso de atención.

Tabla 21.

Conocimientos tecnológicos necesarios para la investigación

N°	Nombre	Aplicación
1	JAVA	Lenguaje de programación usado para el desarrollo en back-end
2	XML	Lenguaje de marcado usado para crear las interfaces en front-end
3	HTML	Lenguaje de marcado que se usó para el diseño de algunos textos dentro del aplicativo
4	ZXING	Librería usada para la gestión de códigos QR

Fuente: Elaboración propia

Teléfono móvil Android fue utilizado debido a que existían ciertas limitaciones con el emulador las cuales eran suplidas por un teléfono físico. Además, el producto final será usado en teléfonos Android físicos, por ello, fue importante probar el aplicativo en un entorno más próximo a la realidad.

SCRUM fue la metodología que se usó para el desarrollo debido a su gran nivel de adaptabilidad. El desarrollo incremental que esta metodología ofrece ayudó a poder medir el avance del proyecto con cada incremento, dando una alerta temprana en caso haya algún

retraso. Además, se hizo uso de un Sprint Backlog priorizado, el cual se encuentra en el

Anexo N° 6.

Tabla 22.

Roles de Scrum en el proyecto

N°	Rol	Responsable	Función
1	Scrum Master	Stefano Tamayo	Apoyar al equipo promoviendo el enfoque al cumplimiento de objetivos
2	Product Owner	Stefano Tamayo	Garantizar que se cumplan con los requerimientos especificados para el proyecto
3	Developer	Stefano Tamayo	Desarrollar el proyecto en base a los requerimientos planteados y recibiendo feedback del PO

Fuente: Elaboración propia

CAPÍTULO IV: DISCUSIÓN Y CONCLUSIONES

4.1. Discusión

Uno de los objetivos específicos de esta investigación fue desarrollar una aplicación móvil para mejorar el tiempo de obtención de datos de un vehículo, en una empresa automotriz. En esta investigación se comprobó la reducción del tiempo de obtención de dichos datos, en concordancia con lo expuesto por (Panduro Lara & Tello Pacheco, 2020) en su tesis, en la cual, al implementar su aplicativo móvil para la gestión de entrada y salida de los colaboradores, se ahorró tiempo simplificando el proceso manual.

Otro objetivo específico fue desarrollar una aplicación móvil para mejorar el tiempo de obtención de datos de los servicios aplicados, en una empresa automotriz. En esta investigación se encontró una reducción considerable en los tiempos de obtención de datos de los servicios aplicados a un vehículo en particular. Asimismo, (Crisostomo Aylas, 2020) señala en su tesis que al implementar una aplicación móvil se mejora la eficiencia en la reducción de tiempos, debido a que el aplicativo móvil influye directamente en los tiempos de atención al cliente.

El tercer objetivo específico fue desarrollar una aplicación móvil para mejorar el tiempo de obtención de datos de los repuestos instalados, en una empresa automotriz. Se cumplió este objetivo, debido a que el análisis demostró que los tiempos de obtención de datos de los repuestos instalados en un vehículo, se redujeron significativamente. De la misma manera, en la tesis de (Huaman Cruz, 2020), se concluyó que existe un impacto positivo entre el tiempo en el cobro de boletos y el uso del aplicativo móvil implementado en su tesis.

Finalmente, el ultimo objetivo específico fue desarrollar una aplicación móvil para mejorar el tiempo de registro de datos de los repuestos y servicios, en una empresa automotriz. Este objetivo también fue cumplido y se demostró la reducción en los tiempos de registro de datos de los repuestos y servicios, tras comparar los tiempos pre-test y post-test. Sucede algo similar en la tesis de (Crisostomo Aylas, 2020), donde después de realizar el comparativo de antes y después de implementar el aplicativo móvil, se concluyó que la reducción de los tiempos fue de 5.26 minutos.

El objetivo general de esta investigación fue desarrollar una aplicación móvil para mejorar la atención al cliente en una empresa del rubro automotriz de la ciudad de Lima, 2022. Se encontró que existe una fuerte relación entre el uso del aplicativo móvil y los tiempos de atención al cliente, debido a que, tras el uso en diferentes escenarios, se manifestó una importante reducción en los tiempos. En concordancia con la investigación de (Huaman Cruz, 2020), donde se comprueba la existencia de un impacto creciente y positivo en la atención al cliente. Su investigación, al igual que esta investigación, han demostrado la relación que tienen con los aplicativos móviles en sus diferentes etapas y en diferentes sectores empresariales. Esto es muy importante para la rentabilidad de la empresa, debido a que como señala (Duque Fierro, Saldarriaga Castrillón, & Bohorquez Ordoñez, 2016) la atención al cliente es el medio por el cual las empresas pueden crecer y permanecer en el mercado.

Uno de los procedimientos necesarios para lograr los objetivos específicos de esta investigación fue modelar y crear las tablas o entidades que conformen la base de datos. Para esta investigación, se usó una base de datos no relacional propia de Firebase; por el contrario, (Dascalu, y otros, 2018) para el desarrollo de su aplicativo móvil usaron una

base de datos relacional en MySQL. Sin embargo, los resultados no se vieron afectados. Al parecer, el tipo de base de datos no afecta directamente al usuario final del aplicativo móvil.

Con el fin de cumplir con los objetivos específicos de esta investigación se desarrolló el front-end y el back-end del aplicativo móvil. El aplicativo móvil de esta investigación fue desarrollado en Android, ya que ofrece una mayor flexibilidad e interfaces amigables. Asimismo, en la investigación de (Gasca Mantilla, Camargo Ariza, & Medina Delgado, 2014) se señala que Android, debido a su característica de edición de archivos xml, permite crear interfaces más amigables para el usuario y sin mucha complejidad. Todo apunta a que Android es una opción muy versátil para el desarrollo de aplicativos móviles.

4.2. Implicancias

Esta investigación deja ciertas consecuencias, partiendo desde el trabajo realizado para el desarrollo del aplicativo móvil. Algunas de esas implicancias son:

Los resultados obtenidos en esta investigación confirman lo que señala (Vergara Vásquez, 2019), en cuya tesis se concluyó la existencia de la relación entre la aplicación móvil y la atención al cliente. En esta investigación, dicha relación también fue comprobada por medio de la reducción de tiempos en los procesos. Esta reducción en el tiempo de espera otorga a los clientes una mejor experiencia y permite a los colaboradores brindarles una atención eficaz.

Esta investigación ayudó a mejorar el proceso y reducir los tiempos de atención al cliente. Brindándole a los colaboradores una mayor facilidad para acceder a los datos, ya que ahora los tienen en la palma de su mano. Y que, a su vez, ellos les brinden una mejor

atención a los clientes, lo cual mejora la experiencia del cliente y por consiguiente aumenta la rentabilidad de la empresa.

Esta investigación mejoró otros 2 puntos muy importantes además del tiempo de atención. Primero, hubo una reducción de esfuerzo por parte de los colaboradores, ya que no requieren desplazarse hasta un centro de datos para solicitar la información. Segundo, los colaboradores son más autónomos y ya no necesitan estar consultando esta información con su supervisor. Adicionalmente, esto permite que el supervisor pueda centrarse en realizar sus propias funciones sin mayores distracciones.

4.3. Limitaciones

Como toda investigación, esta contó con ciertas limitaciones, las cuales son detalladas a continuación:

- No se obtuvo el permiso de alguna empresa, por ello se optó por diseñar una solución genérica y con una base de datos propia, en base a la investigación que se hizo sobre las características de los servicios, repuestos, atención, etc.
- No se cuenta con la totalidad del flujo de la empresa, el alcance de la investigación está definido para ser de ayuda a los colaboradores que atienden los vehículos. Por ello, no se pueden registrar nuevos vehículos, categorías, repuestos o servicios, ya que esto es parte de un flujo distinto.
- El alcance de la investigación no permitió usar la geolocalización como método para identificar el lugar de la sede. Por ello, la sede debe ingresarse de forma manual, al registrar un servicio. Esto significa unos pocos segundos más añadidos al proceso.

- Se presentaron ciertas dificultades y retrasos al momento de obtener las respuestas de la población, debido a la poca disponibilidad de los encuestados. Además, algunos no leyeron correctamente las indicaciones por lo que existen un par de respuestas que se ven erráticas en comparación a las demás.

- Por motivos económicos, no fue posible publicar el aplicativo en Play Store, por lo cual se envió el archivo de extensión apk a los colaboradores para que lo instalen, esto supuso un retraso, ya que para instalar aplicaciones fuera de Play Store se requiere aplicar cierta configuración en el dispositivo objetivo.

4.4. Conclusiones

En concordancia con los objetivos planteados y los hallazgos alcanzados en la investigación, se llegó a las siguientes conclusiones:

Con el uso del aplicativo móvil desarrollado, el tiempo de obtención de datos de los vehículos de los clientes, se redujo de 10.5 minutos a 6.4 minutos, lo cual implica un ahorro de tiempo de aproximadamente un 40% en comparación con el tiempo inicial. Lo cual ha sido el propósito del primer objetivo específico.

En cuanto al segundo objetivo específico, el tiempo de obtención de datos de un servicio aplicado, tuvo una reducción en los tiempos de 10.4 minutos a 6.6 minutos, lo que representa aproximadamente un 37% en la reducción de este tiempo. En ese sentido, el aplicativo móvil desarrollado mejora el tiempo de obtención de datos de un servicio.

Respecto al tercer objetivo específico, podemos concluir que, al usar el aplicativo móvil, el tiempo de obtención de datos de un repuesto instalado, se redujo visiblemente, comparando el tiempo inicial de 10.6 minutos y llegando a reducirlo a 6.1 minutos, alcanzando así un 43% en la reducción de tiempos.

Respecto al cuarto objetivo específico, podemos concluir que, usando el aplicativo móvil, el registro de datos también muestra una reducción en los tiempos, cuyo tiempo promedio inicial era 10.6 minutos y se logró reducir hasta 6.9 minutos, lo que significa una reducción del 35% en el tiempo de esta actividad.

Finalmente, en base a lo concluido en los objetivos específicos, en esta investigación podemos manifestar que los tiempos de atención al cliente antes del aplicativo móvil (pre-test) y después del aplicativo móvil (post-test), se redujeron en aproximadamente entre un 30% y 40% considerando las diferentes consultas o registros que tiene que hacer los colaboradores para una óptima atención. Logrando una mejora en el proceso de atención al cliente (AsIs) hasta llegar al nuevo proceso de atención al cliente (ToBe) y demostrando que la metodología óptima para el desarrollo fue Scrum ya que permitió desarrollar el aplicativo por incrementos e ir mostrando a los usuarios poco a poco el aplicativo móvil, de esta manera se pudieron identificar errores o mejoras antes de tener el producto final.

4.5. Recomendaciones

En base a la experiencia obtenida en la investigación, se dan las siguientes recomendaciones para futuras investigaciones:

- Tener un medio directo de comunicación con los encuestados. Se podría agendar alguna reunión dependiendo la disponibilidad de los encuestados. En la medida de lo posible, solicitar que respondan al cuestionario durante la reunión para evitar que se olviden y así evitar retrasos en los tiempos. Adicionalmente, considerar un periodo de tiempo entre los cuestionarios pre-test y post-test para evitar que los encuestados confundan el objetivo del cuestionario que están respondiendo.

- Intentar adquirir acceso para subir el aplicativo móvil a Google Play Store. En caso de no contar con los recursos económicos, se podría reunir un grupo de desarrolladores para obtener acceso conjunto, esto debe hacerse con tiempo para lograr obtener el acceso. Es recomendable que se suba a Google Play Store, debido a que muchas personas no tienen los conocimientos para instalar archivos apk.

- Al tener la versión final del aplicativo móvil, probarlo en distintos dispositivos de gama baja, media y alta para comprobar que funcione correctamente. Además, diferentes dispositivos móviles tienen diferentes dimensiones, se debería comprobar el orden de los componentes de la interfaz para evitar pérdidas de información y que el usuario final tenga una experiencia óptima.

Referencias

- Baena Paz, G. (2014). *Metodología de la investigación*. Mexico, D.F.: Editorial Patria.
- Crisostomo Aylas, D. J. (2020). *Implementación de una app móvil para mejorar la eficiencia en la reducción del tiempo de recepción del centro de distribución, Huachipa 2020*. Obtenido de Universidad Privada del Norte: <https://hdl.handle.net/11537/30289>
- Cruz, C., Olivares, S., & González, M. (2014). *Metodología de la Investigación*. Mexico: Editorial Patria.
- Dascalu, L. C., Chivu, O., Babiş, C., Iacobescu, G., Dimitrescu, A., & Niculae, E. (2018). *DESIGN AND DEVELOPMENT OF A MOBILE APPLICATION ANDROID FOR THE MEASUREMENT OF STRESS LEVEL*. Obtenido de Fiability & Durability: <https://search.ebscohost.com/login.aspx?direct=true&site=eds-live&db=owf&AN=135968833&authtype=uid&user=rma-browserextension&password=Br0wserExtension789!>
- Diario Gestión. (2014). *Aplicaciones desconectadas: herramientas de competitividad en el mercado*. Obtenido de Gestion.pe: <https://archivo.gestion.pe/empresas/aplicaciones-desconectas-herramientas-competitividad-mercado-2106361>
- Duque Fierro, G. A., Saldarriaga Castrillón, J. F., & Bohorquez Ordoñez, E. (2016). *El servicio al cliente y sus problemáticas actuales: tendencias investigativas*. Obtenido de Revista Criterios,23(1) (p.383-396): <https://revistas.umariana.edu.co/index.php/Criterios/issue/view/137>
- Enrique Villegas, F., Meibol Fajardo, L., & Vásquez, C. (2014). *Gestión de satisfacción al cliente en el sector comercial*. Obtenido de Revista Yachana: <http://revistas.ulvr.edu.ec/index.php/yachana/article/view/22>
- Ferrán, M. (2001). *SPSS para Windows: Analisis Estadístico*. Mexico: McGraw Hill.
- Gasca Mantilla, M. C., Camargo Ariza, L. L., & Medina Delgado, B. (2014). *Metodología para el desarrollo de aplicaciones móviles*. Obtenido de Tecnura: <https://www.redalyc.org/articulo.oa?id=257030546003>
- Hernández Sampieri, R., Fernández Collado, C., & Baptista Lucio, M. d. (2014). *Metodología de la investigación - Sexta Edición*. Mexico: McGraw Hill Education.
- Huaman Cruz, H. F. (2020). *APLICACIÓN MÓVIL GENERADORA DE TICKETS*. Obtenido de Universidad Privada del Norte: <https://hdl.handle.net/11537/25865>
- Khrais, L., & Alghamdi, A. (2021). *The Role of Mobile Application Acceptance in Shaping E-Customer Service*. Obtenido de Future internet: <https://doi.org/10.3390/fi13030077>
- Monsalve Sossa, D. E., Zapata, M. E., Gómez Adasme, M. E., & Ríos, J. P. (2018). *Pruebas a aplicaciones móviles: avances y retos*. Obtenido de Universidad Católica Luis Amigo: <https://doi.org/10.21501/21454086.2983>
- Morales Reyna, F. R., Terranova Borja, G. F., & Sandoya Mayorga, L. (2019). *La calidad en el servicio al cliente: factor de éxito de las empresas*. Obtenido de Opuntia Brava: <https://search.ebscohost.com/login.aspx?direct=true&site=eds-live&db=owf&AN=139663380&authtype=uid&user=rma-browserextension&password=>

- ord=Br0wserExtension789!
- Neri Guzmán, J. C. (2022). *Infraestructura tecnológica en universidades públicas como base para la vinculación con la industria automotriz: el caso de la zona metropolitana de San Luis Potosí*. Obtenido de Revista de Economía: <https://doi.org/10.33937/reveco.2022.249>
- Panduro Lara, C. A., & Tello Pacheco, F. R. (2020). *Implementación de una aplicación móvil de registro de asistencia para reducir el tiempo en el proceso de ingreso y salida del personal en organizaciones en el año 2020*. Obtenido de Universidad Privada del Norte: <https://hdl.handle.net/11537/25145>
- Santamaría Puerto, G., & Hernández Rincón, E. (2015). *Aplicaciones Médicas Móviles: definiciones, beneficios y riesgos*. Obtenido de Salud Uninorte: <https://www.redalyc.org/articulo.oa?id=81745378016>
- Song, Y., Liu, K., Guo, L., Yang, Z., & Jin, M. (2022). *Does hotel customer satisfaction change during the COVID-19? A perspective from online reviews*. Obtenido de ScienceDirect: <https://doi.org/10.1016/j.eswa.2022.116649>
- Valenzuela Salazar, N. L., Buentello Martínez, C. P., Alanis Gomez, L., & Villareal Sánchez, V. (2019). *La atención al cliente, el servicio, el producto y el precio como variables determinantes de la satisfacción del cliente en una pyme de servicios*. Obtenido de Revista Geon, 6(2): <https://doi.org/10.22579/23463910.159>
- van-Velsen, L., JMA-Beaujean, D., & van-Gemert-Pijnen, J. E. (2013). *Why mobile health app overload drives us crazy, and how to restore the sanity*. Obtenido de BMC Medical Informatics and Decision Making: <https://doi.org/10.1186/1472-6947-13-23>
- Vergara Vásquez, J. D. (2019). *USO DE UNA APLICACIÓN MÓVIL Y SU RELACIÓN EN EL PROCESO DE ATENCIÓN AL CLIENTE EN UN RESTAURANTE DE LA CIUDAD DE TRUJILLO*. Obtenido de Universidad Privada del Norte: <https://hdl.handle.net/11537/24066>

Anexos

Anexo N°1. Matriz de operacionalización de la variable

Desarrollo de un aplicativo móvil para mejorar la atención al cliente de una empresa automotriz de la ciudad de Lima, 2022					
VARIABLE DEPENDIENTE	DEFINICIÓN CONCEPTUAL	DEFINICIÓN OPERACIONAL	DIMENSIONES	INDICADORES	ESCALA DE MEDICIÓN
Atención al cliente	La atención al cliente es el medio por el cual las empresas pueden crecer y permanecer en el mercado, todo esto debido a la fidelidad de sus clientes, ya que, sin ellos no habría a quien ofrecerles sus servicios. (Duque Fierro, Saldarriaga Castrillón, & Bohorquez Ordoñez, 2016).	Una buena atención al cliente (variable), es medida a través del tiempo de respuesta que ofrecen los colaboradores al atenderlos. Los indicadores serán los diferentes tiempos durante el proceso de atención.	-Tiempo de atención al cliente	<ul style="list-style-type: none"> •Tiempo de obtención de datos de un vehículo •Tiempo de obtención de los servicios aplicados •Tiempo de obtención de los repuestos instalados •Tiempo de registro de datos 	-Nominal -Ordinal

Anexo N°2. Matriz de consistencia

Desarrollo de un aplicativo móvil para mejorar la atención al cliente de una empresa automotriz de la ciudad de Lima, 2022					
PROBLEMA GENERAL	OBJETIVO GENERAL	HIPÓTESIS GENERAL	VARIABLE DEPENDIENTE	METODOLOGÍA	POBLACIÓN
¿De qué manera el desarrollo de un aplicativo móvil mejora la atención al cliente de una empresa automotriz de la ciudad de Lima, 2022?	Desarrollar una aplicación móvil para mejorar la atención al cliente en una empresa del rubro automotriz de la ciudad de Lima, 2022.	La aplicación móvil mejora la atención al cliente en una empresa del rubro automotriz de la ciudad de Lima, 2022.	Atención al cliente	Enfoque: Cuantitativo Alcance: Explicativo Tipo de investigación: Aplicada Experimental Diseño: Experimental Preexperimental, pre-test / post-test	La población definida en la investigación fueron los colaboradores de cierta empresa, los cuales están encargados de la atención al cliente. Dichos colaboradores son 8, por lo cual no se usará una muestra sino la totalidad de la población
PROBLEMAS ESPECÍFICOS	OBJETIVOS ESPECÍFICOS	HIPÓTESIS ESPECÍFICAS	VARIABLE INDEPENDIENTE	METODOLOGÍA	
<p>¿Cómo un aplicativo móvil puede ayudar en la agilización del proceso de atención al cliente?</p> <p>¿De qué manera mejora la experiencia del cliente usando un aplicativo móvil como apoyo tecnológico a quienes les brindan atención?</p>	<ul style="list-style-type: none"> Desarrollar una aplicación móvil para mejorar el tiempo de obtención de datos de un vehículo, en una empresa automotriz. Desarrollar una aplicación móvil para mejorar el tiempo de obtención de datos de los servicios aplicados, en una empresa automotriz. Desarrollar una aplicación móvil para mejorar el tiempo de obtención de datos de los repuestos instalados, en una empresa automotriz. Desarrollar una aplicación móvil para mejorar el tiempo de registro de datos de los repuestos y servicios, en una empresa automotriz. 	<ul style="list-style-type: none"> La aplicación móvil mejora el tiempo de obtención de datos de un vehículo, en una empresa automotriz. La aplicación móvil mejora el tiempo de obtención de datos de los servicios aplicados, en una empresa automotriz. La aplicación móvil mejora el tiempo de obtención de datos de los repuestos instalados, en una empresa automotriz. La aplicación móvil mejora el tiempo de registro de datos de los repuestos y servicios, en una empresa automotriz. 	Desarrollo de un aplicativo móvil	<p>Técnica: Encuesta Instrumento: Cuestionario</p> <p>Método de análisis de datos: Análisis descriptivo, usando tablas de frecuencias y gráficos estadísticos.</p> <p>Análisis inferencial, por medio de una prueba de normalidad de datos y la aplicación de la prueba de Wilcoxon</p>	

Anexo N°3. Instrumento de recolección de datos

Sección 1 de 2

Formulario de aplicativo Atención al cliente

Descripción del formulario

Aplicativo para ayudarle con la atención al cliente

Hola, me llamo Stefano Tamayo, soy estudiante de decimo ciclo de la carrera de ingeniería de sistemas. El motivo de esta encuesta es conocer la situación actual de los procesos de atención al cliente y así brindarle a usted, como colaborador, una forma más rápida de consultar los datos. Los datos serán usados y comparados para la investigación que estoy realizando.

IMPORTANTE: Por favor, responda con total sinceridad, ya que estos datos servirán para ayudar a **mejorar** el proceso, por lo que no importa si en esta encuesta los tiempos son muy grandes.

Responderé con total sinceridad la siguiente encuesta y tengo conocimiento de que mis respuestas se usarán para una investigación que ayudará a mejorar el proceso actual de atención al cliente.

Sí

No

Sección 2 de 2

Preguntas: (5 preguntas)

Descripción (opcional)

1/5 *

¿Requiere desplazarse a un centro de datos o consultar con otra persona para obtener la información de un vehículo?

Sí

No

2/5

*

¿Cuánto tiempo tarda en promedio en obtener los datos de un vehículo, desde que llega?
(Incluyendo tiempo de desplazamiento a la oficina o el tiempo de respuesta de la persona que consultó)

- aproximadamente 1-2 minutos
- aproximadamente 3-4 minutos
- aproximadamente 5-6 minutos
- aproximadamente 7 minutos
- aproximadamente 8 minutos
- aproximadamente 9 minutos
- aproximadamente 10 minutos
- aproximadamente 11 minutos
- aproximadamente 12 minutos

3/5

*

¿Cuánto tiempo tarda en promedio en obtener los datos de servicios hechos a un vehículo, desde que llega?
(Incluyendo tiempo de desplazamiento a la oficina o el tiempo de respuesta de la persona que consultó)

- aproximadamente 1-2 minutos
- aproximadamente 3-4 minutos
- aproximadamente 5-6 minutos
- aproximadamente 7 minutos
- aproximadamente 8 minutos
- aproximadamente 9 minutos
- aproximadamente 10 minutos
- aproximadamente 11 minutos
- aproximadamente 12 minutos

4/5

*

¿Cuánto tiempo tarda en promedio en obtener los datos de repuestos instalados a un vehículo, desde que llega?

(Incluyendo tiempo de desplazamiento a la oficina o el tiempo de respuesta de la persona que consultó)

- aproximadamente 1-2 minutos
- aproximadamente 3-4 minutos
- aproximadamente 5-6 minutos
- aproximadamente 7 minutos
- aproximadamente 8 minutos
- aproximadamente 9 minutos
- aproximadamente 10 minutos
- aproximadamente 11 minutos
- aproximadamente 12 minutos

5/5

*

¿Cuánto tiempo tarda en promedio en registrar un repuesto o servicio a un vehículo, desde que llega?

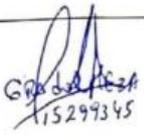
(Incluyendo tiempo de desplazamiento a la oficina o el tiempo de respuesta de la persona que consultó)

- aproximadamente 1-2 minutos
- aproximadamente 3-4 minutos
- aproximadamente 5-6 minutos
- aproximadamente 7 minutos
- aproximadamente 8 minutos
- aproximadamente 9 minutos
- aproximadamente 10 minutos
- aproximadamente 11 minutos
- aproximadamente 12 minutos

Anexo N°4. Matriz de evaluación de experto 1

MATRIZ PARA EVALUACIÓN DE EXPERTOS				
Título de la investigación:	Desarrollo de un aplicativo móvil para mejorar la atención al cliente de una empresa automotriz de la ciudad de Lima, 2022			
Línea de investigación:	Desarrollo sostenible y Gestión empresarial			
Apellidos y nombres del experto:	Guido Trujillo Valdiviezo			
El instrumento de medición pertenece a la variable:	Atención al cliente			
Mediante la matriz de evaluación de expertos, Ud. tiene la facultad de evaluar cada una de las preguntas marcando con una "x" en las columnas de SÍ o NO. Asimismo, le exhortamos en la corrección de los ítems, indicando sus observaciones y/o sugerencias, con la finalidad de mejorar la coherencia de las preguntas sobre la variable en estudio.				
Items	Preguntas	Aprecia		Observaciones
		SÍ	NO	
1	¿El instrumento de medición presenta el diseño adecuado?	x		
2	¿El instrumento de recolección de datos tiene relación con el título de la investigación?	x		
3	¿En el instrumento de recolección de datos se mencionan las variables de investigación?	x		
4	¿El instrumento de recolección de datos facilitará el logro de los objetivos de la investigación?	x		
5	¿El instrumento de recolección de datos se relaciona con las variables de estudio?	x		
6	¿La redacción de las preguntas tienen un sentido coherente y no están sesgadas?	x		
7	¿Cada una de las preguntas del instrumento de medición se relaciona con cada uno de los elementos de los indicadores?	x		
8	¿El diseño del instrumento de medición facilitará el análisis y procesamiento de datos?	x		
9	¿Son entendibles las alternativas de respuesta del instrumento de medición?	x		
10	¿El instrumento de medición será accesible a la población sujeto de estudio?	x		
11	¿El instrumento de medición es claro, preciso y sencillo de responder para, de esta manera, obtener los datos requeridos?	x		
Sugerencias:				
Firma del experto:				
				
Ing. Guido Trujillo Valdiviezo DNI: 25570359				

Anexo N°5. Matriz de evaluación de experto 2

MATRIZ PARA EVALUACIÓN DE EXPERTOS				
Título de la investigación:	Desarrollo de un aplicativo móvil para mejorar la atención al cliente de una empresa automotriz de la ciudad de Lima, 2022			
Línea de investigación:	Desarrollo sostenible y Gestión empresarial			
Apellidos y nombres del experto:	Oscar Emiliano Grados Meza			
El instrumento de medición pertenece a la variable:	Atención al cliente			
<p>Mediante la matriz de evaluación de expertos, Ud. tiene la facultad de evaluar cada una de las preguntas marcando con una "x" en las columnas de SÍ o NO. Asimismo, le exhortamos en la corrección de los ítems, indicando sus observaciones y/o sugerencias, con la finalidad de mejorar la coherencia de las preguntas sobre la variable en estudio.</p>				
Items	Preguntas	Aprecia		Observaciones
		SÍ	NO	
1	¿El instrumento de medición presenta el diseño adecuado?	x		
2	¿El instrumento de recolección de datos tiene relación con el título de la investigación?	x		
3	¿En el instrumento de recolección de datos se mencionan las variables de investigación?	x		
4	¿El instrumento de recolección de datos facilitará el logro de los objetivos de la investigación?	x		
5	¿El instrumento de recolección de datos se relaciona con las variables de estudio?	x		
6	¿La redacción de las preguntas tienen un sentido coherente y no están sesgadas?	x		
7	¿Cada una de las preguntas del instrumento de medición se relaciona con cada uno de los elementos de los indicadores?	x		
8	¿El diseño del instrumento de medición facilitará el análisis y procesamiento de datos?	x		
9	¿Son entendibles las alternativas de respuesta del instrumento de medición?	x		
10	¿El instrumento de medición será accesible a la población sujeto de estudio?	x		
11	¿El instrumento de medición es claro, preciso y sencillo de responder para, de esta manera, obtener los datos requeridos?	x		
<p>Sugerencias:</p>				
<p>Firma del experto:</p> <div style="text-align: center;">  O. Grados Meza 15299345 </div>				

Anexo N°6. Sprint Backlog priorizado

Prioridad	Requerimientos	Sprint estimado
1	Preparar prototipos para el desarrollo del aplicativo móvil	Sprint 1
2	Modelar la base de datos del aplicativo móvil	Sprint 1
3	Crear las tablas o entidades que conforman la Base de Datos	Sprint 1
4	Construir e implementar el módulo de registro	Sprint 2
5	Construir e implementar el módulo para el inicio de sesión	Sprint 2
6	Realizar la integración con Firebase para iniciar sesión con Google	Sprint 2
7	Integrarse con la base de datos de Firebase	Sprint 2
8	Construir e implementar el módulo para consultar los datos de un vehículo	Sprint 3
9	Construir e implementar el módulo para consultar los repuestos de un vehículo	Sprint 3
10	Agregar la funcionalidad de consultar la garantía del repuesto instalado	Sprint 3
11	Construir e implementar el módulo para consultar los servicios aplicados a un vehículo	Sprint 3
12	Construir e implementar el módulo para agregar repuestos	Sprint 4
13	Construir e implementar el módulo para agregar servicios	Sprint 4
14	Implementar el uso de la librería Zxing	Sprint 4
15	Construir e implementar el módulo para generar códigos QR	Sprint 4
16	Agregar la opción de compartir el código QR creado	Sprint 5
17	Implementar el reconocimiento de imágenes para escanear el código QR	Sprint 5
18	Realizar las pruebas, para validar las funcionalidades del aplicativo móvil	Sprint 5

Anexo N°7. Código fuente de ActivityRepuestos.java

```
public class ActivityRepuestos extends AppCompatActivity {

    TextView txtTitulo;
    Button btnRetroceder;
    FloatingActionButton btnNuevo;
    SearchView svBuscar;

    RecyclerView rvDatos;
    private List<Datos> listaDatos= new ArrayList<>();
    private List<Datos> listaDatosAux= new ArrayList<>();
    Adaptador adaptador;

    //globales
    String placa=null;
    String tipo=null;

    public boolean actualizar=false;
    BDFirebase bdFirebase=new BDFirebase(ActivityRepuestos.this);
    Vehiculo vehiculo=new Vehiculo();

    @RequiresApi(api = Build.VERSION_CODES.O)
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_repuestos);
        asignarReferencias();
        cargarDatos();
    }
    private void asignarReferencias() {
        txtTitulo = findViewById(R.id.txtTitulo);
        btnRetroceder = findViewById(R.id.btnRetroceder);
        btnNuevo=findViewById(R.id.btnNuevo);
        rvDatos = findViewById(R.id.rvDatos);
        svBuscar = findViewById(R.id.svBuscar);

        placa=getIntent().getStringExtra("placa");
        tipo=getIntent().getStringExtra("tipo");
        Spanned titulo=null;
        switch (tipo){
            case "repuestos":
                titulo= Html.fromHtml("<p>Repuestos del vehiculo de
placa: <b>"+placa+"</b></p>");
                break;
            case "servicios":
                titulo= Html.fromHtml("<p>Servicios del vehiculo de
placa: <b>"+placa+"</b></p>");
                break;
        }
        txtTitulo.setText(titulo);

        btnNuevo.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
```

```
        Intent intent=new
Intent(ActivityRepuestos.this,Nuevo.class);
        intent.putExtra("tipo",tipo);
        intent.putExtra("placa",placa);
        startActivity(intent);
    }
});

    svBuscar.setOnQueryTextListener(new
SearchView.OnQueryTextListener() {
        @Override
        public boolean onQueryTextSubmit(String query) {
            return false;
        }

        @Override
        public boolean onQueryTextChange(String newText) {
            buscar(newText);
            return true;
        }
    });

    btnRetroceder.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            onBackPressed();
        }
    });
}
@RequiresApi(api = Build.VERSION_CODES.O)
private void cargarDatos() {

    Handler handler = new Handler();
    cargarVehiculo(placa);
    handler.postDelayed(new Runnable() {
        @Override
        public void run() {
            ToDatos();
            cargarListaAux();
        }
    }, 700);
}

    public void buscar(String s){
        listaDatos.clear();
        if (s.equals("")) listaDatos.addAll(listaDatosAux);
        else {
            for (int i = 0; i < listaDatosAux.size(); i++) {
                if
(listaDatosAux.get(i).getNombre().toUpperCase().contains(s.toUpperCase())
) {
                    listaDatos.add(listaDatosAux.get(i));
                }
            }
        }
    }
}
```

```

    }
    adaptador.notifyDataSetChanged();
}

private void cargarVehiculo(String placa){

bdFirebase.getReference().child("Vehiculo").addValueEventListener(new
 ValueEventListener() {
    @Override
    public void onDataChange(@NonNull DataSnapshot snapshot) {
        for (DataSnapshot i: snapshot.getChildren()){
            Vehiculo v=i.getValue(Vehiculo.class);
            if(v.getPlaca().equals(placa))
vehiculo.setVehiculo(v);
        }
    }
    @Override
    public void onCancelled(@NonNull DatabaseError error) {}
});
}
@RequiresApi(api = Build.VERSION_CODES.O)
private List<Datos> ToDatos(){
    listaDatos.clear();

    DateTimeFormatter dtf = DateTimeFormatter.ofPattern("dd/MM/yy
HH:mm");
    String now =dtf.format(LocalDateTime.now());
    LocalDateTime dnow= null;
    dnow = LocalDateTime.parse(now,dtf);

    LocalDateTime drep= null;

    switch (tipo) {
        case "repuestos":
            for (int i = 0; i <
vehiculo.getRepuestoComprados().size(); i++) {
                RepuestoComprado current =
vehiculo.getRepuestoComprados().get(i);
                drep = LocalDateTime.parse(current.getFyh(), dtf);
                Duration datediff = Duration.between(drep, dnow);
                int diffMonth = (int) (datediff.toDays() / 30);

                String estadoG = ((current.getGarantia() - diffMonth
> 0) ? "Activa" : "Vencida");
                listaDatos.add(new Datos(vehiculo.getPlaca(),
current.getRepuesto().getNombre(),
                    "Marca: " + current.getRepuesto().getMarca()
+ "\nCantidad: " + current.getCantidad() +
                    "\nFecha y Hora: " + current.getFyh()
+ "\nTecnico: " + current.getTecnico() +
                    "\nTiempo de garantia: " +
current.getGarantia() + " meses\nEstado de garantia: " + estadoG));
            }
            break;
    }
}

```

```
        case "servicios":
            for (int i = 0; i <
vehiculo.getServicioComprados().size(); i++) {
                ServicioComprado current =
vehiculo.getServicioComprados().get(i);
                listaDatos.add(new Datos(vehiculo.getPlaca(),
current.getServicio().getNombre(),
                    "Tipo de servicio: " +
current.getTipoServicio() + "\nFecha y Hora: " + current.getFyh() +
                    "\nTecnico: " + current.getTecnico()
+ "\nSede: " + current.getSede()));
            }
            break;
        }
        return listaDatos;
    }
    private void cargarListaAux(){
        listaDatosAux.addAll(listaDatos);
        adaptador=new Adaptador(ActivityRepuestos.this, listaDatos);
        rvDatos.setAdapter(adaptador);
        rvDatos.setLayoutManager(new
LinearLayoutManager(ActivityRepuestos.this));
    }
}
```

Anexo N°8. Código fuente de Adaptador.java

```
public class Adaptador extends
RecyclerView.Adapter<Adaptador.MyViewHolder>{

    //Consultar upn_firebase
    private Context context;
    List<Datos> listaDatos=new ArrayList<>();
    Drawable imgWarranty=Drawable.createFromPath("garantia.png");

    public Adaptador(Context context, List<Datos> listaDatos){
        this.context=context;
        this.listaDatos=listaDatos;
    }

    @NonNull
    @Override
    public MyViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int
viewType) {
        LayoutInflater inflater=LayoutInflater.from(this.context);
        View vista = inflater.inflate(R.layout.fila,parent,false);
        return new MyViewHolder(vista);
    }

    @Override
    public void onBindViewHolder(@NonNull MyViewHolder holder, int
position) {
        holder.filaNombre.setText(listaDatos.get(position).getNombre());
        String desc=listaDatos.get(position).getDescripcion();
        String[] partDesc=desc.split("Estado de garantia: ");
        holder.filaDescripcion.setText(partDesc[0]);

        if (partDesc.length>1) {
            System.out.println(partDesc[1]);
            if (partDesc[1].equals("Activa"))
                holder.filaImagen.setImageResource(R.drawable.garantia);
            else
                holder.filaImagen.setImageResource(R.drawable.garantia2);
        }
        else
            holder.filaImagen.setVisibility(View.GONE);
    }

    @Override
    public int getItemCount() {
        return listaDatos.size();
    }

    public class MyViewHolder extends RecyclerView.ViewHolder {
        TextView filaNombre, filaDescripcion;
        LinearLayout fila;
        ImageView filaImagen;
        public MyViewHolder(@NonNull View itemView) {
            super(itemView);
            filaNombre=itemView.findViewById(R.id.filaNombre);
        }
    }
}
```

```
        filaDescripcion=itemView.findViewById(R.id.filaDescripcion);  
        filaImagen=itemView.findViewById(R.id.filaImagen);  
        fila=itemView.findViewById(R.id.fila);  
    }  
}
```

Anexo N°9. Código fuente de Consultar_Placa.java

```
public class Consultar_Placa extends AppCompatActivity {

    Button btnRetroceder, btnBuscar, btnRepuestos, btnServicios;
    EditText txtPlaca;
    TextView lblResultados;
    String globalPlaca;
    Vehiculo vehiculo=new Vehiculo();

    //Firebase
    BDFirebase bdFirebase=new BDFirebase(Consultar_Placa.this);

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_consultar_placa);
        asignarReferencias();
        inicializar();
    }
    private void inicializar(){
        btnRepuestos.setVisibility(View.GONE);
        btnServicios.setVisibility(View.GONE);
        if (getIntent().hasExtra("scannedPlaca")){
            txtPlaca.setText(getIntent().getStringExtra("scannedPlaca"));
            Handler handler = new Handler();
            handler.postDelayed(() -> btnBuscar.performClick(), 300);
        }
    }
    private void asignarReferencias() {
        btnRetroceder = findViewById(R.id.btnRetroceder);
        btnRepuestos = findViewById(R.id.btnRepuestos);
        btnServicios = findViewById(R.id.btnServicios);
        btnBuscar = findViewById(R.id.btnBuscar);
        txtPlaca = findViewById(R.id.txtPlaca);
        lblResultados = findViewById(R.id.lblResultados);

        btnBuscar.setOnClickListener(new View.OnClickListener() {
            @RequiresApi(api = Build.VERSION_CODES.N)
            @Override
            public void onClick(View v) {
                vehiculo=new Vehiculo();
                String placa=txtPlaca.getText().toString();

                cargarVehiculo(placa);
                Handler handler = new Handler();
                handler.postDelayed(() -> cargarInfo(), 700);

                hideSoftKeyboard();
            }
        });
        btnRepuestos.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
```

```

        Intent intent=new
Intent (Consultar_Placa.this,ActivityRepuestos.class);
        intent.putExtra("tipo","repuestos");
        intent.putExtra("placa",globalPlaca);
        startActivity(intent);
    }
});
btnServicios.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent intent=new
Intent (Consultar_Placa.this,ActivityRepuestos.class);
        intent.putExtra("tipo","servicios");
        intent.putExtra("placa",globalPlaca);
        startActivity(intent);
    }
});
btnRetroceder.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        onBackPressed();
    }
});
}
private void hideSoftKeyboard(){
    if(getCurrentFocus()!=null && getCurrentFocus() instanceof
EditText){
        InputMethodManager imm =
        (InputMethodManager) getSystemService (Context.INPUT_METHOD_SERVICE);
        imm.hideSoftInputFromWindow(getWindow().getCurrentFocus().getWindowToken(
), 0);
    }
}

@RequiresApi(api = Build.VERSION_CODES.N)
private void cargarInfo(){

    Spanned resultados = null;
    String placa=txtPlaca.getText().toString();

    if (vehiculo.getPlaca()!=null){
        globalPlaca=placa;
        resultados= Html.fromHtml("<h3>Datos del vehiculo: </h3><br>"
+
                "<p><b>Placa:"
</b>"+vehiculo.getPlaca()+"</p>"+
                "<p><b>VIN: </b>"+vehiculo.getVin()+"</p>"+
                "<p><b>Dueño:"
</b>"+vehiculo.getDueno()+"</p>"+
                "<p><b>Modelo:"
</b>"+vehiculo.getModelo()+"</p>"+
                "<p><b>Peso Bruto:"
</b>"+vehiculo.getPeso_bruto()+" Kg.</p>"+

```

```
                "<p><b>Kilometraje:  
</b>" + vehiculo.getKilometraje() + " Km.</p>" +  
                "<p><b>Horas: </b>" + vehiculo.getHoras() + "  
hrs.</p>",  
                Html.FROM_HTML_MODE_COMPACT);  
        btnRepuestos.setVisibility(View.VISIBLE);  
        btnServicios.setVisibility(View.VISIBLE);  
    }  
    else {  
        resultados = Html.fromHtml("<p>La placa <b>" + placa + "</b> no ha  
sido encontrada</p>",  
                Html.FROM_HTML_MODE_COMPACT);  
        btnRepuestos.setVisibility(View.GONE);  
        btnServicios.setVisibility(View.GONE);  
    }  
    lblResultados.setText(resultados);  
}  
private void cargarVehiculo(String placa) {  
    bdFirestore.getReference().child("Vehiculo").addValueEventListener(new  
ValueEventListener() {  
        @Override  
        public void onDataChange(@NonNull DataSnapshot snapshot) {  
            for (DataSnapshot i: snapshot.getChildren()) {  
                Vehiculo v = i.getValue(Vehiculo.class);  
                if (v.getPlaca().equals(placa))  
                    vehiculo.setVehiculo(v);  
            }  
        }  
        @Override  
        public void onCancelled(@NonNull DatabaseError error) {}  
    });  
}
```

Anexo N°10. Código fuente de EscanearQr.java

```
public class EscanearQr extends AppCompatActivity {

    Button btnRetroceder, btnEscanear;
    TextView lblError;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_escanear_qr);
        asignarReferencias();
    }

    private void asignarReferencias(){
        btnRetroceder = findViewById(R.id.btnRetroceder);
        btnEscanear = findViewById(R.id.btnEscanear);
        lblError = findViewById(R.id.lblError);

        lblError.setText("");

        btnEscanear.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                scanQr();
            }
        });

        btnRetroceder.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                onBackPressed();
            }
        });
    }

    private void scanQr(){
        ScanOptions options = new ScanOptions();
        options.setDesiredBarcodeFormats(ScanOptions.QR_CODE);
        options.setPrompt("Escanee el QR del cliente");
        options.setCameraId(0);
        options.setBeepEnabled(true);
        options.setBarcodeImageEnabled(true);
        options.setOrientationLocked(true);
        options.setCaptureActivity(Capturar.class);
        barLauncher.launch(options);
    }
    ActivityResultLauncher<ScanOptions> barLauncher =
    registerForActivityResult(new ScanContract(), result ->
    {
        if(result.getContents() != null){
            Pattern pattern = Pattern.compile("[A-Z0-9]{3}-[A-Z0-9]{3}$",
            Pattern.CASE_INSENSITIVE);
            Matcher matcher = pattern.matcher(result.getContents());
            boolean matchFound = matcher.find();
        }
    });
}
```

```
        Toast.makeText(this, result.getContents(),
Toast.LENGTH_SHORT).show();

        if(matchFound) {
            Intent intent = new Intent(EscanearQr.this,
Consultar_Placa.class);
            intent.putExtra("scannedPlaca", result.getContents());
            startActivity(intent);
            lblError.setText("");
        }
        else {
            lblError.setText("El código Qr escaneado no corresponde a
un placa");
        }
    }
    });
}
```

Anexo N°11. Código fuente de GenerarQr.java

```

public class GenerarQr extends AppCompatActivity {

    Button btnGenerar, btnRetroceder, btnCompartir;
    TextView lblCreado;
    EditText txtPlaca;
    ImageView imgQr;
    Bitmap bitmap;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_generar_qr);
        asignarReferencias();
        inicializar();
    }

    private void asignarReferencias(){
        btnRetroceder = findViewById(R.id.btnRetroceder);
        btnGenerar = findViewById(R.id.btnGenerar);
        btnCompartir = findViewById(R.id.btnCompartir);
        lblCreado = findViewById(R.id.lblCreado);
        txtPlaca = findViewById(R.id.txtPlaca);
        imgQr = findViewById(R.id.imgQr);

        btnGenerar.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Pattern pattern = Pattern.compile("[A-Z0-9]{3}-[A-Z0-9]{3}$", Pattern.CASE_INSENSITIVE);
                Matcher matcher =
                pattern.matcher(txtPlaca.getText().toString());
                boolean matchFound = matcher.find();
                hideSoftKeyboard();
                try {
                    BarcodeEncoder barcodeEncoder = new BarcodeEncoder();

                    bitmap=barcodeEncoder.encodeBitmap(txtPlaca.getText().toString(),
                    BarcodeFormat.QR_CODE,750,750);
                    if (matchFound){
                        imgQr.setImageBitmap(bitmap);
                        lblCreado.setText("Su código QR ha sido generado
con éxito\n\nComparta este codigo con el cliente dueño del vehículo");
                        lblCreado.setVisibility(View.VISIBLE);
                        btnCompartir.setVisibility(View.VISIBLE);
                    }
                    else {
                        imgQr.setImageBitmap(null);
                        lblCreado.setText("El texto ingresado debe ser
una placa");
                        lblCreado.setVisibility(View.VISIBLE);
                    }
                } catch (WriterException e) {
                    e.printStackTrace();
                }
            }
        });
    }
}

```

```

    }
  });

  btnCompartir.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
      Drawable drawable=imgQr.getDrawable();
      Bitmap bm=((BitmapDrawable)drawable).getBitmap();

      try {
        File file = new
File(getApplicationContext().getExternalCacheDir(), File.separator
+"ImageDemo.jpg");
        FileOutputStream fOut = new FileOutputStream(file);
        bm.compress(Bitmap.CompressFormat.JPEG, 100, fOut);
        fOut.flush();
        fOut.close();
        file.setReadable(true, false);
        final Intent intent = new
Intent(android.content.Intent.ACTION_SEND);
        intent.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
        Uri photoURI =
FileProvider.getUriForFile(getApplicationContext(),
BuildConfig.APPLICATION_ID + ".provider", file);

        intent.putExtra(Intent.EXTRA_STREAM, photoURI);

intent.addFlags(Intent.FLAG_GRANT_READ_URI_PERMISSION);
        intent.setType("image/jpeg");

        startActivity(Intent.createChooser(intent, "Share
image via"));
      } catch (Exception e) {
        e.printStackTrace();
      }
    }
  });

  btnRetroceder.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
      onBackPressed();
    }
  });
}

public static Bitmap viewToBitmap(View view, int width, int height){
  Bitmap
bitmap=Bitmap.createBitmap(width,height,Bitmap.Config.ARGB_8888);
  Canvas canvas=new Canvas(bitmap);
  view.draw(canvas);
  return bitmap;
}

```

```
private void inicializar() {
    lblCreado.setVisibility(View.GONE);
    btnCompartir.setVisibility(View.GONE);
}
private void hideSoftKeyboard() {
    if (getCurrentFocus() != null && getCurrentFocus() instanceof
EditText) {
        InputMethodManager imm =
(InputMethodManager) getSystemService(Context.INPUT_METHOD_SERVICE);

imm.hideSoftInputFromWindow(getWindow().getCurrentFocus().getWindowToken(
), 0);
    }
}
}
```

Anexo N°12. Código fuente de Home.java

```
public class Home extends AppCompatActivity {

    Button btnCerrarSesion,btnConsultarPlaca,btnEscanearQr,btnGenerarQr;
    TextView lblBienvenido;

    String name;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_home);
        asignarReferencias();
    }
    @Override
    public void onBackPressed () {
    }
    private void asignarReferencias() {
        SharedPreferences preferences=
        getSharedPreferences(getString(R.string.prefs_file),
        Context.MODE_PRIVATE);
        name=preferences.getString("name", null);

        btnCerrarSesion = findViewById(R.id.btnCerrarSesion);
        btnConsultarPlaca = findViewById(R.id.btnConsultarPlaca);
        btnEscanearQr = findViewById(R.id.btnEscanearQr);
        btnGenerarQr = findViewById(R.id.btnGenerarQr);
        lblBienvenido = findViewById(R.id.lblBienvenido);

        lblBienvenido.setText("Bienvenido (a), "+name);
        btnCerrarSesion.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                AlertDialog ynDialog = new AlertDialog.Builder(Home.this)
                    .setTitle("Cerrando sesión...")
                    .setMessage("¿Está seguro que desea cerrar
sesión?")
                    .setPositiveButton("Aceptar",null)
                    .setNegativeButton("Cancelar",null)
                    .show();
                Button
                pButton=ynDialog.getButton(AlertDialog.BUTTON_POSITIVE);
                pButton.setOnClickListener(new View.OnClickListener() {
                    @Override
                    public void onClick(View v) {
                        SharedPreferences.Editor preferences=
                        getSharedPreferences(getString(R.string.prefs_file),
                        Context.MODE_PRIVATE).edit();
                        preferences.clear();
                        preferences.apply();
                        Intent intent = new Intent(Home.this,
MainActivity.class);
                        startActivity(intent);
                    }
                }
            }
        }
    }
}
```

```
        });  
    }  
});  
btnConsultarPlaca.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        Intent intent =new  
Intent(Home.this,Consultar_Placa.class);  
        startActivity(intent);  
    }  
});  
btnGenerarQr.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        Intent intent =new Intent(Home.this,GenerarQr.class);  
        startActivity(intent);  
    }  
});  
btnEscanearQr.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        Intent intent =new Intent(Home.this,EscanearQr.class);  
        startActivity(intent);  
    }  
});  
}  
}
```

Anexo N°13. Código fuente de MainActivity.java

```
public class MainActivity extends AppCompatActivity {

    Button btnIniciarSesion, btnRegistrare,btnGoogle;
    EditText txtCorreo, txtClave;

    private static final int GOOGLE_SIGN_IN = 100;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        asignarReferencias();
        sesion();
    }

    private void asignarReferencias() {
        btnIniciarSesion=findViewById(R.id.btnIniciarSesion);
        btnRegistrare=findViewById(R.id.btnRegistrare);
        btnGoogle=findViewById(R.id.btnGoogle);

        txtCorreo=findViewById(R.id.txtCorreo);
        txtClave=findViewById(R.id.txtClave);

        btnIniciarSesion.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String correo, clave;
                correo = txtCorreo.getText().toString();
                clave = txtClave.getText().toString();
                Intent intent = new Intent(MainActivity.this,
Home.class);
                if (valCredenciales(correo,clave)){

                    FirebaseAuth.getInstance().signInWithEmailAndPassword(correo,
clave).addOnCompleteListener(command -> {
                        if (command.isSuccessful()) {
                            capturarSesion();
                            Toast.makeText(MainActivity.this, "Bienvenido",
Toast.LENGTH_SHORT).show();
                            startActivity(intent);
                        } else {
                            Toast.makeText(MainActivity.this, "Su correo y/o
clave son incorrectos", Toast.LENGTH_SHORT).show();
                        }
                    });
                } else {
                    Toast.makeText(MainActivity.this, "Su correo y/o clave
son incorrectos", Toast.LENGTH_SHORT).show();
                }
            });
        });
        btnGoogle.setOnClickListener(new View.OnClickListener() {
            @Override
```

```

        public void onClick(View v) {
            GoogleSignInOptions googleConf = new
GoogleSignInOptions.Builder(GoogleSignInOptions.DEFAULT_SIGN_IN).requestI
dToken(getString(R.string.default_web_client_id)).requestEmail().build();

            GoogleSignInClient googleClient=
GoogleSignIn.getClient(MainActivity.this,googleConf);
            googleClient.signOut();

startActivityForResult(googleClient.getSignInIntent(),GOOGLE_SIGN_IN);
        }
    });
    btnRegistrare.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            Intent intent=new
Intent(MainActivity.this,Registro.class);
            startActivity(intent);
        }
    });
}
private void capturarsesion(){
    SharedPreferences.Editor preferences=
getSharedPreferences(getString(R.string.prefs_file),
Context.MODE_PRIVATE).edit();
    preferences.putString("correo",txtCorreo.getText().toString());

preferences.putString("name",txtCorreo.getText().toString().substring(0,7
));
    preferences.apply();
}
private void capturarsesionG(String cor,String nam,String lNam){
    SharedPreferences.Editor preferences=
getSharedPreferences(getString(R.string.prefs_file),
Context.MODE_PRIVATE).edit();
    preferences.putString("correo",cor);
    preferences.putString("name",nam);
    preferences.putString("lastName",lNam);
    preferences.apply();
}
private void sesion(){
    SharedPreferences preferences=
getSharedPreferences(getString(R.string.prefs_file),
Context.MODE_PRIVATE);
    String sscorreopreferences=preferences.getString("correo",null);
    System.out.println("GOT IT");
    System.out.println(sscorreopreferences);
    if (sscorreopreferences!=null){
        if(!sscorreopreferences.equals("")) {
            Intent intent = new Intent(MainActivity.this,
Home.class);
            startActivity(intent);
            System.out.println("Aaaaaaaa" + sscorreopreferences);
        }
    }
}

```

```

    }
}

@Override
protected void onActivityResult(int requestCode, int resultCode,
@Nullable Intent data) {
    super.onActivityResult(requestCode, resultCode, data);

    if(requestCode==GOOGLE_SIGN_IN){
        Task<GoogleSignInAccount> task =
        GoogleSignIn.getSignedInAccountFromIntent(data);

        try {

            GoogleSignInAccount account = task.getResult();

            if (account != null) {
                AuthCredential credential =
                GoogleAuthProvider.getCredential(account.getIdToken(), null);

                FirebaseAuth.getInstance().signInWithCredential(credential).addOnCompleteListener(command -> {
                    if (command.isSuccessful()) {

                        capturarsesionG(account.getEmail(),account.getGivenName(),account.getFamilyName());

                        Toast.makeText (MainActivity.this,
                        "Bienvenido", Toast.LENGTH_SHORT).show();
                        Intent intent = new Intent(MainActivity.this,
                        Home.class);

                        intent.putExtra("name",account.getGivenName());
                        startActivity(intent);
                    } else {
                        Toast.makeText (MainActivity.this, "Error
                        login Google", Toast.LENGTH_SHORT).show();
                    }
                });
            }
        } catch (Exception exception) {
            Toast.makeText (MainActivity.this, "Error login Google",
            Toast.LENGTH_SHORT).show();
        }
    }
}

private boolean valCredenciales(String correo, String pass){
    if (regex(correo, "^[^@]+@[^@]+\\.[a-zA-Z]{2,}$")&&pass.length()>6)
        return true;
    else
        return false;
}

```

```
    }  
    public boolean regex(String texto, String patron) {  
        Pattern pattern = Pattern.compile(patron,  
Pattern.CASE_INSENSITIVE);  
        Matcher matcher = pattern.matcher(texto);  
        boolean matchFound = matcher.find();  
        return matchFound;  
    }  
}
```

Anexo N°14. Código fuente de Nuevo.java

```
public class Nuevo extends AppCompatActivity {

    Button btnRetroceder, btnGuardar;
    TextView txtTitulo, txtDescripcion;
    Spinner spinnerRS, spinnerRS2;

    ArrayList<Categoria> categoriaListR=new ArrayList<>();
    ArrayList<Categoria> categoriaListS=new ArrayList<>();
    ArrayList<Repuesto> listaTodosRepuestos=new ArrayList<>();
    ArrayList<Servicio> listaTodosServicios=new ArrayList<>();

    ArrayList<Repuesto> listaRepuestos=new ArrayList<>();
    ArrayList<Servicio> listaServicios=new ArrayList<>();

    Repuesto repuesto=new Repuesto();
    Servicio servicio=new Servicio();

    Spanned titulo=null;
    String textoBoton=null;
    String prompt=null;
    String prompt2=null;
    String confirmTitle=null;
    Spanned confirmMessage=null;
    String confirmGuardado=null;

    Vehiculo vehiculo=new Vehiculo();
    //globales
    String placa=null;
    String tipo=null;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_nuevo);
        asignarReferencias();
    }

    private void asignarReferencias() {
        txtTitulo = findViewById(R.id.txtTitulo);
        txtDescripcion = findViewById(R.id.txtDescripcion);
        btnRetroceder = findViewById(R.id.btnRetroceder);
        btnGuardar=findViewById(R.id.btnGuardar);
        spinnerRS=findViewById(R.id.spinnerRS);
        spinnerRS2=findViewById(R.id.spinnerRS2);

        placa=getIntent().getStringExtra("placa");
        tipo=getIntent().getStringExtra("tipo");
        switch (tipo){
            case "repuestos":
                esRepuesto();
                break;
            case "servicios":
                esServicio();
        }
    }
}
```

```

        break;
    }
    txtTitulo.setText(titulo);
    btnGuardar.setText(textoBoton);

    btnGuardar.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            AlertDialog confirm = new AlertDialog.Builder(Nuevo.this)
                .setTitle(confirmTitle)
                .setMessage(confirmMessage)
                .setPositiveButton("Aceptar", new
DialogInterface.OnClickListener() {
                    @RequiresApi(api = Build.VERSION_CODES.O)
                    @Override
                    public void onClick(DialogInterface dialog,
int which) {
                        ultimosDatos();
                        cargarVehiculo(placa);
                    }
                })
                .setNegativeButton("Cancelar", null)
                .show();
            Button
pButton=confirm.getButton(AlertDialog.BUTTON_POSITIVE);
        }
    });
    btnRetroceder.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            onBackPressed();
        }
    });
}
private void cargarSpinner() {
}
private void esRepuesto() {
    titulo= Html.fromHtml("<p>Nuevo repuesto del vehiculo de placa:
<b>"+placa+"</b></p>");
    textoBoton="REGISTRAR REPUESTO";

    prompt="Seleccione categoria";
    prompt2="Seleccione repuesto";
    Repuesto selectedRepuesto=new Repuesto();

    //DUMMY
    categoriaListR.clear();
    categoriaListR.add(new Categoria("0",prompt));
    cargarCategoriasR();
    listaRepuestos.add(new Repuesto("0",new
Categoria("0","k"),prompt2,"","",""));
    //DUMMY

```

```

        ArrayAdapter<Categoria> adapterRep=new ArrayAdapter<>(this,
androidx.appcompat.R.layout.support_simple_spinner_dropdown_item,categori
aListR);
        spinnerRS.setAdapter(adapterRep);
        spinnerRS.setPrompt(prompt);
        spinnerRS.setOnItemSelectedListener(new
AdapterView.OnItemSelectedListener() {
            @Override
            public void onItemSelected(AdapterView<?> parent, View view,
int position, long id) {
                Toast.makeText(Nuevo.this,
categoriaListR.get(position).getNombre(), Toast.LENGTH_SHORT).show();
                listaRepuestos.clear();
                spinnerRS2.setSelection(0);
                txtDescripcion.setText("");
                listaRepuestos.add(new Repuesto("0",new
Categoria("0","k"),prompt2,"",""));

                listaRepuestos.addAll(filtrarRepuestos(categoriaListR.get(position).getId
()));
                listaTodosRepuestos.clear();
                System.out.println("Selected Categoria"+listaRepuestos);
            }

            @Override
            public void onNothingSelected(AdapterView<?> parent) {

            }
        });

        ArrayAdapter<Repuesto> adapterRep2=new ArrayAdapter<>(this,
androidx.appcompat.R.layout.support_simple_spinner_dropdown_item,listaRep
uestos);
        spinnerRS2.setAdapter(adapterRep2);
        spinnerRS2.setPrompt(prompt2);
        spinnerRS2.setOnItemSelectedListener(new
AdapterView.OnItemSelectedListener() {
            @Override
            public void onItemSelected(AdapterView<?> parent, View view,
int position, long id) {
                Toast.makeText(Nuevo.this,
listaRepuestos.get(position).getNombre(), Toast.LENGTH_SHORT).show();
                if(position!=0) {
                    String sku = listaRepuestos.get(position).getSku();
                    String nombre =
listaRepuestos.get(position).getNombre();
                    String marca =
listaRepuestos.get(position).getMarca();
                    String descripcion =
listaRepuestos.get(position).getDescripcion();
                    Spanned desc = null;
                    desc = Html.fromHtml("<h5 style=\"text-
align:center\">\" + nombre + "</h5>\" +

```

```

        "<p>SKU: " + sku + "<br>" +
        "Marca: " + marca + "</p><p>" + descripcion +
"</p>");
        txtDescripcion.setVisibility(View.VISIBLE);
        btnGuardar.setVisibility(View.VISIBLE);
        txtDescripcion.setText(desc);

selectedRepuesto.setRepuesto(listaRepuestos.get(position));
        repuesto.setRepuesto(selectedRepuesto);
        confirmMessage=Html.fromHtml("<p><b>Repuesto:
</b>"+selectedRepuesto.getNombre()+"</p>" +
        "<p><b>Placa: </b>"+placa+"</p>");
    }
    else {
        txtDescripcion.setVisibility(View.INVISIBLE);
        btnGuardar.setVisibility(View.INVISIBLE);
    }
    System.out.println("Selected Repuesto"+listaRepuestos);
}

@Override
public void onNothingSelected(AdapterView<?> parent) {
}

});

confirmTitle="Confirmar repuesto: ";
confirmGuardado="Repuesto guardado";
}
private void esServicio(){
    titulo= Html.fromHtml("<p>Nuevo servicio del vehiculo de placa:
<b>"+placa+"</b></p>");
    textoBoton="REGISTRAR SERVICIO";
    prompt="Seleccione categoria";
    prompt2="Seleccione servicio";

    Servicio selectedServicio=new Servicio();

    //DUMMY
    categoriaListS.clear();
    categoriaListS.add(new Categoria("0",prompt));
    cargarCategoriasS();
    listaServicios.add(new Servicio("0",new
Categoria("0", "0"),prompt2,""));
    //DUMMY

    ArrayAdapter<Categoria> adapterRep=new ArrayAdapter<>(this,
androidx.appcompat.R.layout.support_simple_spinner_dropdown_item,categori
aListS);
    spinnerRS.setAdapter(adapterRep);
    spinnerRS.setPrompt(prompt);
    spinnerRS.setOnItemSelectedListener(new
AdapterView.OnItemSelectedListener() {

```

```

        @Override
        public void onItemSelected(AdapterView<?> parent, View view,
int position, long id) {
            Toast.makeText (Nuevo.this,
categorialistS.get (position).getNombre (), Toast.LENGTH_SHORT).show ();
            listaServicios.clear ();
            spinnerRS2.setSelection (0);
            txtDescripcion.setText ("");
            listaServicios.add (new Servicio ("0", new
Categoria ("0", "0"), prompt2, ""));

listaServicios.addAll (filtrarServicios (categorialistS.get (position).getId
()));
            listaTodosServicios.clear ();
            System.out.println ("Selected Categoria"+listaServicios);
        }

        @Override
        public void onNothingSelected(AdapterView<?> parent) {

        }
    });

    ArrayAdapter<Servicio> adapterRep2=new ArrayAdapter<>(this,
androidx.appcompat.R.layout.support_simple_spinner_dropdown_item, listaSer
vicios);
    spinnerRS2.setAdapter (adapterRep2);
    spinnerRS2.setPrompt (prompt2);
    spinnerRS2.setOnItemSelectedListener (new
AdapterView.OnItemSelectedListener () {
        @Override
        public void onItemSelected(AdapterView<?> parent, View view,
int position, long id) {
            Toast.makeText (Nuevo.this,
listaServicios.get (position).getNombre (), Toast.LENGTH_SHORT).show ();
            if (position!=0) {
                String nombre =
listaServicios.get (position).getNombre ();
                String descripcion =
listaServicios.get (position).getDescripcion ();
                Spanned desc = null;
                desc = Html.fromHtml ("

##### <b>Servicio: </b>"+selectedServicio.getNombre ()+"</p>" +


```

```

        "<p><b>Placa: </b>"+placa+"</p>");
    }
    else {
        txtDescripcion.setVisibility(View.INVISIBLE);
        btnGuardar.setVisibility(View.INVISIBLE);
    }
    System.out.println("Selected Servicio"+listaServicios);
}

@Override
public void onNothingSelected(AdapterView<?> parent) {

}

});

confirmTitle="Confirmar servicio: ";
confirmGuardado="Servicio guardado";
}

BDFirebase bdFirebase=new BDFirebase(Nuevo.this);
List<Categoria> lista=new ArrayList<>();
List<Repuesto> listaR=new ArrayList<>();
List<Servicio> listaS=new ArrayList<>();
private void cargarCategoriasR(){
    lista.clear();

bdFirebase.getReference().child("Categoria_Repuestos").addValueEventListener(new ValueEventListener() {
    @Override
    public void onDataChange(@NonNull DataSnapshot snapshot) {
        for (DataSnapshot i: snapshot.getChildren()){
            Categoria c = i.getValue(Categoria.class);
            lista.add(c);
        }
        System.out.println(lista);
        categoriaListR.addAll(lista);
    }

    @Override
    public void onCancelled(@NonNull DatabaseError error) {

    }

});
}
private void cargarCategoriasS(){
    lista.clear();

bdFirebase.getReference().child("Categoria_Servicios").addValueEventListener(new ValueEventListener() {
    @Override
    public void onDataChange(@NonNull DataSnapshot snapshot) {
        for (DataSnapshot i: snapshot.getChildren()){
            Categoria c = i.getValue(Categoria.class);
            lista.add(c);
        }
    }
});
}
}

```

```
    }
    System.out.println(lista);
    categoriaListS.addAll(lista);
}

@Override
public void onCancelled(@NonNull DatabaseError error) {
}
});
}
private void cargarRepuestos(){
    listaR.clear();

    bdFirestore.getReference().child("Repuestos").addValueEventListener(new
    ValueEventListener() {
        @Override
        public void onDataChange(@NonNull DataSnapshot snapshot) {
            for (DataSnapshot i: snapshot.getChildren()){
                Repuesto c = i.getValue(Repuesto.class);
                listaR.add(c);
            }
            System.out.println(listaR);
            listaTodosRepuestos.addAll(listaR);
        }

        @Override
        public void onCancelled(@NonNull DatabaseError error) {
        }
    });
}
private void cargarServicios(){
    listaS.clear();

    bdFirestore.getReference().child("Servicios").addValueEventListener(new
    ValueEventListener() {
        @Override
        public void onDataChange(@NonNull DataSnapshot snapshot) {
            for (DataSnapshot i: snapshot.getChildren()){
                Servicio c = i.getValue(Servicio.class);
                listaS.add(c);
            }
            System.out.println(listaS);
            listaTodosServicios.addAll(listaS);
        }

        @Override
        public void onCancelled(@NonNull DatabaseError error) {
        }
    });
}
```

```

public List<Repuesto> filtrarRepuestos(String id){
    cargarRepuestos();
    System.out.println("Todos Repuestos: "+listaTodosRepuestos);
    List<Repuesto> rL=new ArrayList<>();
    for (Repuesto i: listaTodosRepuestos){
        if(i.getCategoria().getId().equals(id)){
            rL.add(i);
            System.out.println("XXXXXXXXXX: "+i.getNombre());
        }
    }
    return rL;
}

public List<Servicio> filtrarServicios(String id){
    cargarServicios();
    List<Servicio> sL=new ArrayList<>();
    for (Servicio i: listaTodosServicios){
        if(i.getCategoria().getId().equals(id)){
            sL.add(i);
            System.out.println("XXXXXXXXXX: "+i.getNombre());
        }
    }
    return sL;
}

@RequiresApi(api = Build.VERSION_CODES.O)
public void registrarCompra(String dato1, String dato2){
    SharedPreferences preferences=
    getSharedPreferences(getString(R.string.prefs_file),
    Context.MODE_PRIVATE);

    DateTimeFormatter dtf;
    dtf = DateTimeFormatter.ofPattern("dd/MM/yy HH:mm");
    String now =dtf.format(LocalDateTime.now());

    String fullName=preferences.getString("name",null)+"
    "+preferences.getString("lastName","");

    switch (tipo){
        case "repuestos":
            RepuestoComprado rc=new
            RepuestoComprado(UUID.randomUUID().toString(), repuesto, Integer.parseInt(d
            ato1), now, fullName, Integer.parseInt(dato2));

            bdFirestore.getReference().child("Repuestos_Comprados").child(rc.getId()).
            setValue(rc);
            vehiculo.getRepuestoComprados().add(rc);
            break;
        case "servicios":
            ServicioComprado sc=new
            ServicioComprado(UUID.randomUUID().toString(), servicio, dato1, now, fullName
            , dato2);

```

```
bdFirebase.getReference().child("Servicios_Comprados").child(sc.getId()).
setValue(sc);
        vehiculo.getServicioComprados().add(sc);
        break;
    }

bdFirebase.getReference().child("Vehiculo").child(vehiculo.getId()).setVa
lue(vehiculo);

    }
    private void cargarVehiculo(String placa){

bdFirebase.getReference().child("Vehiculo").addValueEventListener(new
 ValueEventListener() {
    @Override
    public void onDataChange(@NonNull DataSnapshot snapshot) {
        for (DataSnapshot i: snapshot.getChildren()){
            Vehiculo v=i.getValue(Vehiculo.class);
            if (v.getPlaca().equals(placa))
vehiculo.setVehiculo(v);
        }
    }
    @Override
    public void onCancelled(@NonNull DatabaseError error) {}
});
    }

@RequiresApi(api = Build.VERSION_CODES.O)
private void ultimosDatos(){
    String hint1="", hint2="";
    int inputT=0;

    switch (tipo){
        case "repuestos":
            hint1="Ingrese cantidad";
            hint2="Ingrese garantia (meses)";
            inputT=InputType.TYPE_CLASS_NUMBER;
            break;
        case "servicios":
            hint1="Ingrese tipo de servicio";
            hint2="Ingrese sede del servicio";
            inputT=InputType.TYPE_CLASS_TEXT;
            break;
    }

    final EditText editText2=new EditText(Nuevo.this);
    final EditText editText1=new EditText(Nuevo.this);

    editText2.setInputType(inputT);
    editText2.setHint(hint2);
    AlertDialog confirm2 = new AlertDialog.Builder(Nuevo.this)
        .setTitle("Ingrese estos ultimos datos")
```

```
        .setMessage(hint2)
        .setView(editText2)
        .setPositiveButton("Aceptar", (dialog, which) -> {
            Handler handler = new Handler();
            handler.postDelayed(() ->
registrarResultado(editText1.getText().toString(), editText2.getText().toStri
ng()), 700);

Toast.makeText(Nuevo.this, confirmGuardado, Toast.LENGTH_LONG).show();
        onBackPressed();

    })
    .setNegativeButton("Cancelar", null)
    .show();

editText1.setInputType(inputT);
editText1.setHint(hint1);
AlertDialog confirm1 = new AlertDialog.Builder(Nuevo.this)
    .setTitle("Ingrese estos ultimos datos")
    .setMessage(hint1)
    .setView(editText1)
    .setPositiveButton("Aceptar", null)
    .setNegativeButton("Cancelar", null)
    .show();
    }
}
```

Anexo N°15. Código fuente de Registro.java

```
public class Registro extends AppCompatActivity {

    Button btnVolverIniciarSesion, btnRegistrar;
    EditText txtCorreo, txtClave, txtPass;

    boolean cvalido, pvalido;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_registro);
        asignarReferencias();
    }
    private void asignarReferencias() {
        btnVolverIniciarSesion =
findViewById(R.id.btnVolverIniciarSesion);
        btnRegistrar = findViewById(R.id.btnRegistrar);

        txtCorreo = findViewById(R.id.txtCorreo);
        txtClave = findViewById(R.id.txtClave);
        txtPass = findViewById(R.id.txtPass);

        btnRegistrar.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String correo, clave, pass;
                correo = txtCorreo.getText().toString();
                clave = txtClave.getText().toString();
                pass = txtPass.getText().toString();
                Intent intent = new Intent(Registro.this,
MainActivity.class);
                if (regex(correo, "^[^@]+@[^@]+\\.\\.[a-zA-Z]{2,}$"))
cvalido = true;
                else cvalido = false;
                if (pass.equals(clave)) pvalido=true;
                else pvalido=false;

                if (pvalido && cvalido) {

FirebaseAuth.getInstance().createUserWithEmailAndPassword(correo,
clave).addOnCompleteListener(command -> {
                    if (command.isSuccessful()) {
                        Toast.makeText(Registro.this, "Registro
exitoso", Toast.LENGTH_SHORT).show();
                        startActivity(intent);
                    } else {
                        Toast.makeText(Registro.this, "Error al
registrar", Toast.LENGTH_SHORT).show();
                    }
                });
                }
            }
        });
    }
    else {
        if (!cvalido) Toast.makeText(Registro.this, "Correo
invalido", Toast.LENGTH_SHORT).show();
    }
}
```

```
                if (!pvalido) Toast.makeText(Registro.this, "Claves no  
coinciden", Toast.LENGTH_SHORT).show();  
            }  
        }  
    });  
    btnVolverIniciarSesion.setOnClickListener(new  
View.OnClickListener() {  
        @Override  
        public void onClick(View v) {  
            Intent intent = new Intent(Registro.this,  
MainActivity.class);  
            startActivity(intent);  
        }  
    });  
}  
public boolean regex(String texto, String patron) {  
    Pattern pattern = Pattern.compile(patron,  
Pattern.CASE_INSENSITIVE);  
    Matcher matcher = pattern.matcher(texto);  
    boolean matchFound = matcher.find();  
    return matchFound;  
}  
}
```