

**Desarrollo de Aplicación móvil para la gestión de domicilios en la Lechonería Don
Chancho**

Director(es)
Prof. Dianalin Neme Prada

Diego Alejandro Granados Avendaño
Mayo 2021

Universidad Antonio Nariño.
Facultad de Ingeniería de Sistemas.
Especialización En Ingeniería De Software

Tabla de Contenido

1	Planteamiento del Problema.....	6
1.1	Descripción del problema.....	6
1.2	Formulación del problema	7
1.3	Objetivos	7
1.3.1	Objetivo General.....	7
1.3.2	Objetivos Específicos	7
2	Marco Referencia.....	7
2.1	Estado del arte	7
2.1.1	Trabajos de investigación relacionados con la idea de proyecto.....	8
2.2	Impacto	11
2.3	Componente de innovación.....	12
2.4	Marco Teórico	13
2.4.1	Firestore	13
2.4.2	Flutter	13
2.4.3	Marco Legal.....	14
3	Metodología	14
4	Proceso de Software	15
4.1	Requerimientos funcionales.....	15
4.1.1	Modelo de componentes.....	18
4.2	Requerimientos no funcionales.....	19
4.3	Diseño y arquitectura	24
4.3.1	Diagrama de despliegue	25
4.3.2	Caso de uso Arquitecturalmente relevante	26
4.3.3	Diagrama de Secuencia	27
4.3.4	Diagrama de clases.....	28
4.3.5	Arquitectura de alto nivel	29
4.4	Construcción.....	29
4.4.1	Diseño de la interfaz gráfica.....	30
4.4.2	Construcción y codificación	35
4.5	Pruebas	37
4.5.1	Prueba Inicio de Sesión	37
4.5.2	Prueba Validación Interfaz Rol Administrador.....	39
4.5.3	Prueba automatización de pruebas con BrowserStack	42
4.6	Instalación y Configuración	45
4.6.1	Pasos para la Instalación de la Aplicación de gestión de Domicilios	46
4.7	Conclusiones.....	52
5	Referencias.....	53

Lista de tablas

<i>Tabla 1. Trabajos de investigación relacionados con la idea de proyecto</i>	<i>10</i>
<i>Tabla 2. Impacto Stakeholder.....</i>	<i>12</i>
<i>Tabla 3. Tablero Kanban.....</i>	<i>15</i>
<i>Tabla 4. Requisito funcional Acceso.....</i>	<i>16</i>
<i>Tabla 5. Requisito funcional Definición de Interfaz por cada Rol.....</i>	<i>16</i>
<i>Tabla 6. Requisito funcional Cargue de pedido</i>	<i>17</i>
<i>Tabla 7. Requisito funcional Definición de Interfaz por cada Rol.....</i>	<i>17</i>
<i>Tabla 8. Requisito funcional Búsqueda datos clientes frecuentes.....</i>	<i>17</i>
<i>Tabla 9. Requisito funcional Definición de Interfaz por cada Rol.....</i>	<i>18</i>
<i>Tabla 10. Rendimiento- Especificación Árbol Utilidad.....</i>	<i>20</i>
<i>Tabla 11. Disponibilidad- Especificación Árbol Utilidad.....</i>	<i>21</i>
<i>Tabla 12. Seguridad - Especificación Árbol Utilidad</i>	<i>21</i>
<i>Tabla 13. Escalabilidad - Escenarios de Calidad</i>	<i>22</i>
<i>Tabla 14. Nuevos Productos - Escenarios de Calidad</i>	<i>22</i>
<i>Tabla 15. Seguridad - Escenarios de Calidad.....</i>	<i>23</i>
<i>Tabla 16. Seguridad Administrador- Escenarios de Calidad.....</i>	<i>23</i>
<i>Tabla 17. Seguridad - Escenarios de Calidad.....</i>	<i>24</i>
<i>Tabla 18. Prueba funcional – Inicio de sección.</i>	<i>37</i>
<i>Tabla 19. Prueba Toma de pedidos desde el rol administrador.....</i>	<i>39</i>
<i>Tabla 20. Prueba Datos del cliente</i>	<i>40</i>
<i>Tabla 21. Prueba toma de Pedido</i>	<i>41</i>
<i>Tabla 22. Prueba Confirmación del pedido- parte 1.....</i>	<i>41</i>
<i>Tabla 23. Prueba Confirmación del pedido- parte 2.....</i>	<i>42</i>
<i>Tabla 24. Instalación y Configuración primer y segundo paso</i>	<i>46</i>
<i>Tabla 25. Instalación y Configuración tercer y cuarto paso.....</i>	<i>47</i>
<i>Tabla 26. Instalación y Configuración quinto paso</i>	<i>48</i>
<i>Tabla 27. Instalación y Configuración sexto y séptimo paso</i>	<i>49</i>

Tabla 28. Instalación y Configuración Octavo Paso..... 50

Lista de Ilustraciones

<i>Ilustración 1. Componentes de la aplicación gestión de domicilios en lechonerías</i>	<i>19</i>
<i>Ilustración 2. Árbol de utilidad Resumido.....</i>	<i>19</i>
<i>Ilustración 3. Diagrama de Despliegue.....</i>	<i>25</i>
<i>Ilustración 4. Casos de Uso.....</i>	<i>26</i>
<i>Ilustración 5. Diagrama de Secuencia.....</i>	<i>27</i>
<i>Ilustración 6. Diagrama de contexto</i>	<i>29</i>
<i>Ilustración 7. Selección de Colores y fuente.....</i>	<i>30</i>
<i>Ilustración 8. Registro, recuperar contraseña, crear nueva contraseña.....</i>	<i>30</i>
<i>Ilustración 9. Interfaz administrador.....</i>	<i>31</i>
<i>Ilustración 10. Interfaz Inicio Vendedor(a).....</i>	<i>31</i>
<i>Ilustración 11. Menú común estado de pedidos - Administrador y vendedor</i>	<i>32</i>
<i>Ilustración 12. Editar producto</i>	<i>32</i>
<i>Ilustración 13. Interfaz toma de pedidos</i>	<i>33</i>
<i>Ilustración 14. Interfaz Inicio Motorizado.....</i>	<i>33</i>
<i>Ilustración 15. Interfaz para seguimiento de pedidos-Rol Motorizado.....</i>	<i>34</i>
<i>Ilustración 16. Interfaz Cocinero.....</i>	<i>34</i>
<i>Ilustración 17. Arquitectura en Capas con el patrón – Block de Flutter</i>	<i>35</i>
<i>Ilustración 18. Desglose la arquitectura en capas utilizada en el proyecto</i>	<i>36</i>
<i>Ilustración 19. Prueba correo y Contraseña correctos, Selección Rol equivocado.....</i>	<i>38</i>
<i>Ilustración 20. Prueba campos vacíos.....</i>	<i>38</i>
<i>Ilustración 21. Prueba correo y Contraseña correctos, Selección Rol equivocado.....</i>	<i>39</i>
<i>Ilustración 22. Configuración de entorno Appium.....</i>	<i>43</i>
<i>Ilustración 23. Pruebas realizadas en la web BrowserStack</i>	<i>43</i>
<i>Ilustración 24. Prueba fallida de automatización</i>	<i>44</i>
<i>Ilustración 25. Prueba de ejecución en BrowserStack.....</i>	<i>44</i>
<i>Ilustración 26. Consumo de recurso de la app.....</i>	<i>45</i>

1 Planteamiento del Problema

1.1 Descripción del problema

En la industria de las lechonerías de la calle 27 sur de la ciudad de Bogotá, debido al pequeño tamaño de los locales comerciales, no es posible tener un computador, que permita llevar un control sistematizado de los pedidos; lo que ha generado diferentes problemáticas como: olvido de pedidos, errores humanos en la gestión de domicilio, pérdida de los talonarios, compra de materia prima en exceso, incumplimiento en la hora de entrega, etc.; debido a la problemática anteriormente mencionada, el propietario de la Lechonería Don Chanco se desea el desarrollo de una aplicación móvil para toma de pedidos y gestión de domicilios.

Por tal razón el presente trabajo de grado tendrá como objetivo el desarrollo de una aplicación móvil en la plataforma Android, que permita realizar una toma de pedidos que se ajuste a las características particulares de la Lechonería Don Chanco, ofreciendo seguridad y organización; permitiendo llevar una trazabilidad exacta de cada pedido, por último, se requiere que en esta aplicación se pueda realizar: seguimiento del pedido, requerimiento de materia prima, confirmación del pedido al cliente, alertas de pedidos pendientes, así mismo con las siguientes características:

- El desarrollo permitirá generar una lista de Pedidos Entregados por motorizado.
- Base de Datos Clientes
- Sistema diseñado con una arquitectura escalable que permita la implementación de nuevas funcionalidades.
- La aplicación deberá permitir manejar varios puntos de venta
- Información real del estado del pedido.
- El sistema permitirá confirmar el pedido mediante correo electrónico
- Total, Ventas por Periodo

- La información de pedido se actualizará en todos los roles para que se pueda gestionar en tiempo real.

1.2 Formulación del problema

¿Cómo desarrollar una aplicación bajo el framework Flutter para Android, que permita la gestión de pedidos en lechonerías de la 27 sur en la ciudad de Bogotá?

1.3 Objetivos

1.3.1 Objetivo General

Construir una aplicación móvil para la gestión de domicilios de lechona en la ciudad de Bogotá (Caso de estudio Lechonería Don Chancho).

1.3.2 Objetivos Específicos

- Definir la línea base de arquitectura lógica y tecnológica, para la gestión de pedidos en lechonerías en Bogotá.
- Construir una aplicación móvil que cumpla con las necesidades de gestión de pedidos en lechonerías.
- Aplicar escenarios de calidad, con el fin de evaluar el cumplimiento de necesidades del usuario.

2 Marco Referencia

2.1 Estado del arte

De acuerdo con las publicaciones de varios autores, se puede afirmar que la lechona tuvo sus orígenes en Arabia Saudita y fue adoptada por la cocina española en la tierra de Castilla,

siendo el cochinitillo de Segovia junto con el de Arévalo, uno de los platos más populares en España, donde tiempo después fue traída por los españoles conquistadores alrededor del siglo XVI¹; así mismo se estableció que era utilizada por los españoles para los festejos de ocasiones especiales y en época de navidad como sinónimo de unión familiar; a través de los años, fue adoptada por las familias colombianas, principalmente en el departamento del Tolima, donde se ha hecho famosa la preparación de este plato.

Aunque en otras regiones, como en Bogotá, la lechona se prepara con arroz, y otros tipos de carne, lleva arepa paisa y el tradicional cuero crocante de color dorado. En el departamento del Tolima, no lleva arroz, solo se cocina con arvejas fritas y carne de cerdo que se adoba con cebolla larga, cebolla cabezona, ajo y el toque secreto.

El negocio de las lechonerías en la 27 sur se caracteriza por producir y vender lechona, que es un plato típico de la región Tolima y es considerada de origen español. En la actualidad son cerca de 40 locales que se dedican a la venta de lechona en la llamada "L" ubicada en el Olaya (Bogotá), siendo una alternativa de negocio para las familias que derivan el sustento de esta actividad económica.

2.1.1 Trabajos de investigación relacionados con la idea de proyecto

A continuación, se describen las ventajas y desventajas de 4 aplicaciones relacionadas con el objetivo de este proyecto de grado de las innumerables aplicaciones móviles que hay en el mercado actualmente que ofrecen el servicio de gestión de pedidos móvil; por último, cabe aclarar que actualmente no existe ninguna aplicación de toma de pedidos que cumpla con la dinámica comercial de las lechonerías.

¹ Ximena Muñoz, consultado (22-03-2021), Lechonería Charrasco Link de referencia [tps://sites.google.com/site/lechoneriacharrasco/noticias/lahistoriadelalechona](https://sites.google.com/site/lechoneriacharrasco/noticias/lahistoriadelalechona)

Nombre	Ventajas	Desventajas
<p>Aplicativo móvil en la gestión de pedidos de servicios en WC Solutions Engineering S.A.S.</p>	<ul style="list-style-type: none"> • Envío promoción de los servicios ofrecidos. • Mejora la comunicación con los clientes. • Falta de facilidad de pagos en línea • Visualizar el detalle de un servicio 	<ul style="list-style-type: none"> • Interfaz Poco Amigable • No se encuentra información sobre el "Demo" de la aplicación
<p>Aplicativo móvil para la recepción, solicitud y gestión de domicilios en las tiendas de barrio de Bogotá</p>	<ul style="list-style-type: none"> • Conecta a tenderos con sus clientes • optimiza el servicio ya existente de domicilios de las tiendas de barrio. • Permite a los tenderos modelizar sus negocios. • Optimiza los procesos de los tenderos 	<ul style="list-style-type: none"> • Solo para solicitud de domicilios en Tiendas • Solo es un prototipo funcional de la aplicación móvil de domicilios • Falta de pruebas de Funcionalidad
<p>Software de gestión de pedidos de clientes</p>	<ul style="list-style-type: none"> • Preparar la visita comercial con esta información además de los reportes comerciales. • Llevar siempre contigo (en tu Tablet o IPAD) el catálogo de productos completo. • Aplicar tarifas y promociones sin miedo a equivocarse. • Enviar presupuestos y pedidos al momento. • Impresión de pedidos 	<ul style="list-style-type: none"> • Costo Alto. • Soporte desde España. • Capacidad limitada par nuevos requerimientos específicos. • Poca accesibilidad desde Colombia.

Nombre	Ventajas	Desventajas
<p>Desarrollo de una aplicación móvil de pedido de comida a domicilio para sistemas Android y iOS mediante React-Native</p>	<ul style="list-style-type: none"> • Presencia en dos sistemas operativos predominantes del mercado. • Proporciona una comodidad al usuario a la hora de realizar pedidos de comida. • Sistema intuitivo que agilice el proceso de pedido para el usuario. • Ofrece un sistema de reparto a restaurantes sin los suficientes recursos. 	<ul style="list-style-type: none"> • Los usuarios tienen una los restaurantes con servicio de reparto a domicilio. • Es exclusivo para usuarios finales • No está especializado en el proceso de cada restaurante • Se necesita de registro de los clientes
<p>Desarrollo de Aplicación móvil para toma de pedidos y gestión de domicilios en lechonerías</p>	<ul style="list-style-type: none"> • Integración con el proceso de toma de pedidos en Lechonerías. • Disminuyó tiempo de confirmación del pedido • Interfaz gráfica • Seguimiento de Estado de Pedidos • Bajo Costo de adquisición • Soporte • Escalabilidad 	<ul style="list-style-type: none"> • Desarrollo para solo Android • Difícil aceptación • Requiere de conexión a internet

Tabla 1. Trabajos de investigación relacionados con la idea de proyecto

2.2 Impacto

El presente desarrollo móvil, permitirá establecer una comunicación fluida entre los Stakeholder de la Lechonería Don Chanco, para lo cual se establecieron 4 Módulos con información específica de cada rol, de acuerdo con lo anterior se pretende impactar de manera directa el servicio a los clientes, cumpliendo la promesa que tradicionalmente se ha hecho en las lechonerías que es: hora de entrega, frescura y calidad de la lechona, así mismo se desea impactar en el seguimiento y solución de novedades o problemas en el día a día de las entregas a domicilio, por último se quiere cambiar la percepción que se tiene acerca de las aplicaciones tecnológicas en este sector de la economía, con un diseño IU (User Interface), sencillo, amigable que fomente el interés de los usuarios como una herramienta para posicionar la marca a través de la mejora del servicio al cliente.

En la siguiente tabla se pueden observar los impactos, que se establecieron de acuerdo con las entrevistas de los Stakeholder de la Lechonería Don Chanco.

Stakeholder	Impacto
Propietario Lechonería	<ul style="list-style-type: none"> • Reducir costos en todas las áreas de proceso. • Disminuir las pérdidas por Ordenes sin entregar.
Administrador	<ul style="list-style-type: none"> • Disminuir las quejas de los clientes por retardos en los pedidos y olvido de ordenes de pedido por parte de los subalternos. • Realizar en tiempo real el seguimiento a cada pedido.
Área de Ventas	<ul style="list-style-type: none"> • Mejorar las ventas. • Incremento en las comisiones.
Mensajeros Motorizados	<ul style="list-style-type: none"> • Mejorar la asignación de los pedidos • Horarios de entregas por orden cronológico.

Stakeholder	Impacto
Área de Proceso	<ul style="list-style-type: none"> • Reducción de pérdidas en la materia prima • Realización de pedidos de manera exacta. • Lista de Pedidos con fechas de entrega precisas • Lista de materia prima por fecha

Tabla 2. Impacto Stakeholder

2.3 Componente de innovación

En la actividad comercial de las lechonerías, para el caso de estudio de Don Chanco, donde actualmente no cuenta con una solución tecnológica para la gestión de pedidos, sí bien es cierto que actualmente hay diversas soluciones que permiten gestionar pedidos sin importar el tipo de industria, en la presente investigación no se pudo evidenciar una aplicación diseñada, para cumplir con los requerimientos propios de la gestión de domicilios en lechonerías, ajustándose su dinámica comercial; por lo tanto se pretende innovar acercando la Lechonería Don Chanco con el desarrollo de una aplicación de gestión de domicilios orientada a dispositivos móviles, con las siguientes características: toma de pedido en solo tres pasos, confirmación de pedido al cliente, actualización nueva orden de pedido en los 4 roles principales establecidos en el actual caso de estudio, requerimiento de materia prima basada en la cantidad de ordenes confirmadas y no en aproximaciones, diseño basado en el Core del negocio, todo lo anterior sin conocimientos técnicos para utilizarlo; por lo tanto con la anterior innovación, se pretende mejorar el servicio al cliente mitigando los errores derivados de la gestión tradicional con la que se lleva actualmente la trazabilidad de las ordenes de pedido brindando un mejor servicio.

2.4 Marco Teórico

2.4.1 Firebase

Es una plataforma en la nube para desarrollo de aplicaciones web y móvil, disponible para diferentes plataformas como: iOS, Android y web, siendo creada en 2010² por James Tamplin y Andrew Lee, la cual fue comprada por Google en 2014, caracterizándose como una base de datos en tiempo real, pero actualmente posee varias funciones adicionales, entre ellas hacer más sencilla la creación de aplicaciones webs o móviles, siendo interesante para el actual proyecto, al no tener que dedicarle mucho tiempo al desarrollo backend o al mantenimiento de la bases de datos.

Entre las Funciones de Firebase encontramos *Desarrollo* que ofrece todos los servicios necesarios para el desarrollo del proyecto sin importar si es móvil o web; gracias a las bases de datos “*Realtime database*”, que son alojadas en la nube como No SQL(No relacional) o como datos en formato JSON, así mismo también se destaca la Autenticación de usuarios permitiendo tanto el registro mediante email y contraseña, como el acceso utilizando otras plataformas por ejemplo, Google, Facebook o Twitter y gracias su capa gratuita permitió realizar el proyecto sin pasar a la versión de pago, que pudo haber sido un inconveniente, sin embargo como el actual proyecto se encuentra las primeras etapas, la versión gratuita de Firebase fue suficiente.

2.4.2 Flutter

Flutter es un SDK desarrollado por Google para crear aplicaciones móviles, sin importar si el desarrollo es para Android o iOS de Apple, convirtiéndolo en uno de los proyectos de desarrollo de aplicaciones móviles de más rápido crecimiento, influenciado por la característica principal que ofrece de ser multiplataforma creando interfaces amigables, basadas en widgets, Material Design para Android y Cupertino Style en iOS; los cuales

² Wikipedia, Firebase, consultado (25-04-2021), Link <https://es.wikipedia.org/wiki/Firebase>

esta preconstruídos para lograr un desarrollo más rápido, sin perder la calidad para el usuario final, Además, Flutter posee una característica llamada Hot Reload (en caliente), permitiendo hacer cambios sin necesidad de esperar a recompilar todo el código, todo lo anterior solo son una de las múltiples ventajas por la que se eligió este SDK Flutter, como base para el presente desarrollo.

2.4.3 Marco Legal

Los Clientes podrán ejercer su derecho de hábeas data en cualquier momento y los datos personales almacenados en Firebase, sólo podrán usarse para el propósito original, como lo establece los decretos el Decreto 1377 de 2013, que reglamentó parcialmente la Ley General de Protección de Datos Personales, Ley 1581 de 2012, por lo tanto, se pondrá a disposición el enlace <https://donchancho.com/Politica-de-Datos/>, donde los clientes pueden solicitar la eliminación de sus datos personales de las bases de datos.

3 Metodología

Para el presente trabajo de grado se eligió la metodología Kanban que tiene sus orígenes en el japonés y significa (tarjeta visual)³, el objetivo de este método consiste en establecer un flujo de trabajo constante y organizado; para la actual investigación la Metodología Kanban ayuda a tener una gestión de trabajo fluida de manera integral en el desarrollo de las diferentes tareas gracias a la visualización en las tarjetas (Por hacer, En proceso, Hecho), logrando evitar la sobrecarga de trabajo y estimado el tiempo en el cual se deberían completar las tareas, manteniendo los avances del proyecto en metas alcanzables, lo que garantiza flujo de trabajo estable y así mismo realizar un seguimiento del tiempo gastado en cada tarea y ajustar cambios en las tareas en caso de ser necesario, siempre teniendo en cuenta el objetivo específico del presente trabajo; también podemos destacar que Kanban permite que todas las tareas sean entendibles, facilitando detectar cualquier eventualidad

3 Wikipedia, Kanban, (30-04-2021), Link de consulta <https://es.wikipedia.org/wiki/Kanban>

que se pueda llegar a suceder durante el proceso de desarrollo de alguna de las tareas; por ultimo cabe destacar que se eligió esta metodología por las siguientes características: mejora tu productividad, trabajo organizado, Tiempos de entrega flexibles, asignación de tareas puntuales, calidad en las entregas, baja acumulación de trabajo, Flexibilidad.

To-do	En progreso		Done
	Trabajando	En espera	
Entrenamiento al personal seleccionado de la aplicación Verificación de cumplimiento de los objetivos propuestos		Probar Funcionalidades Pruebas de Aceptación Pruebas en usuarios nuevos Enviar el archivo SDK al Cliente	Levantamiento de información y Análisis de requerimientos de la aplicación Recolectar información por cada área de proceso Revisión y depuración de la información Selección de información relevante para la aplicación Orientar la aplicación a los objetivos del proyecto Marco Teorico Formulación del problema Justificación Metodología Desarrollo del Proyecto (Aplicación de la Ingeniería de Software, con la metodología escogida y cada una de sus etapas) Conclusiones y recomendaciones Diseño de estructura de la aplicación Diseño de interface gráfica de la aplicación Desarrollo de diagrama de flujos y procesos Establecer módulos de aplicación Desarrollo de aplicación en lenguaje de programación establecido Pruebas en celulares Android

Tabla 3. Tablero Kanban

4 Proceso de Software

4.1 Requerimientos funcionales

Un requisito funcional⁴ define la función del desarrollo de la aplicación móvil de gestión de pedidos y de cada uno de sus componentes; esta función es descrita como un conjunto de entradas, comportamientos y salidas. Los requisitos funcionales pueden ser: cálculos, detalles técnicos, manipulación de datos y otras funcionalidades específicas que se supone, un sistema debe cumplir. Los requisitos de comportamiento para cada requisito funcional se muestran en los casos de uso. Son complementados por los requisitos no funcionales, que se enfocan en cambio en el diseño o la implementación.

⁴ Wikipedia, Requisito funcional, (30-04-2021), Link de consulta https://es.wikipedia.org/wiki/Requisito_funcional

Requerimiento	Control de Acceso y Autenticación
Funcionalidad	La aplicación registra y autentica el rol del usuario
Objetivos	<ul style="list-style-type: none"> • Determinar la identidad de cada usuario cada • Definir privilegios de cada Rol
Requerimientos Funcionales	Registro y autenticación de usuarios

Tabla 4. Requisito funcional Acceso

Requerimiento	Definir Interfaz
Funcionalidad	La aplicación define e implementa interfaz para cada Rol: Administrador, Vendedor, Motorizado y Cocinero
Objetivos	<ul style="list-style-type: none"> • Desplegar la interfaz adecuada para cada Rol • Definir privilegios de cada Rol
Requerimientos Funcionales	Despliegue de interfaz para Administrador Despliegue de interfaz para Vendedor Despliegue de interfaz para Motorizado Despliegue de interfaz para Cocinero

Tabla 5. Requisito funcional Definición de Interfaz por cada Rol

Requerimiento	Cargue de pedido
Funcionalidad	La aplicación permite la toma y programación de la orden de pedido
Objetivos	<ul style="list-style-type: none"> • Permitir la digitalización de la orden de pedido • Mejorar el tiempo de toma del pedido • Simplificar en 3 pasos la toma de pedidos
Requerimientos Funcionales	<ul style="list-style-type: none"> • Carga de número de teléfono del cliente • Programación del pedido

	<ul style="list-style-type: none"> • Guardad nueva orden de pedido
--	---

Tabla 6. Requisito funcional Cargue de pedido

Requerimiento	Cargue de pedido
Funcionalidad	La aplicación permite la toma y programación de la orden de pedido
Objetivos	<ul style="list-style-type: none"> • Permitir la digitalización de la orden de pedido • Mejorar el tiempo de toma del pedido • Simplificar en 3 pasos la toma de pedidos
Requerimientos Funcionales	<ul style="list-style-type: none"> • Carga de número de teléfono del cliente • Programación del pedido • Guardad nueva orden de pedido

Tabla 7. Requisito funcional Definición de Interfaz por cada Rol

Requerimiento	Búsqueda de cliente en la base de datos
Funcionalidad	La aplicación busca y diligencia los datos del cliente, si está registrado.
Objetivos	<ul style="list-style-type: none"> • Agilizar la toma de datos • Mejorar el tiempo en la toma del pedido • Simplificar el registro de clientes frecuentes
Requerimientos Funcionales	<ul style="list-style-type: none"> • Administrador o Vendedor digita el número de teléfono del cliente • Programación del pedido • Guardad nueva orden de pedido

Tabla 8. Requisito funcional Búsqueda datos clientes frecuentes

Requerimiento	Confirmación de pedido al cliente vía correo electrónico
Funcionalidad	Enviar confirmación con el detalle del pedido al correo del cliente

Objetivos	<ul style="list-style-type: none"> Mejorar la percepción del cliente de la empresa Don Chanco. Facilitar el seguimiento mediante la confirmación de hora y fecha del pedido al cliente.
Requerimientos Funcionales	<ul style="list-style-type: none"> Administrador o Vendedor deben confirmar la orden de pedido

Tabla 9. Requisito funcional Definición de Interfaz por cada Rol

4.1.1 Modelo de componentes

		<p>Familia Modulo(X) <u>C&C ()</u> Estilo Arquitectural: <u>N-Tier</u></p> <p>Nivel</p> <div style="border: 1px solid black; width: 100px; height: 20px; margin: 5px 0;"></div> <p>Nota</p> <p style="text-align: center;">↓ use</p> <p>Dependencia o Uso</p>	
Título: Componentes de la aplicación	ID: 01	Nivel de Profundidad: 0	Nomenclatura: Libre
Arquitectura: Desarrollador	Grupo: Desarrollador	Fecha: febrero 2021	Versión: 1.0

Ilustración 1. Componentes de la aplicación gestión de domicilios en lechonerías

4.2 Requerimientos no funcionales

Los Requisitos No Funcionales⁵ o atributo de calidad es un requisito que especifica criterios que pueden usarse para evaluar la operación de una aplicación en lugar de sus comportamientos específicos, que normalmente son abordados por los requisitos funcionales; por lo tanto, los RNF son las restricciones o condiciones que impone el producto owner, por ejemplo: Lenguaje de desarrollo y cantidad de usuarios que podría llegar tener la aplicación de manera recurrente.

A continuación, se describe el desarrollo de requisitos no funcionales que se evaluaron para el desarrollo de la aplicación móvil de gestión de pedidos, con la finalidad de lograr RNF que aborden la utilidad de la aplicación móvil, desde los escenarios de: Desempeño, Modificabilidad, Disponibilidad y Seguridad.

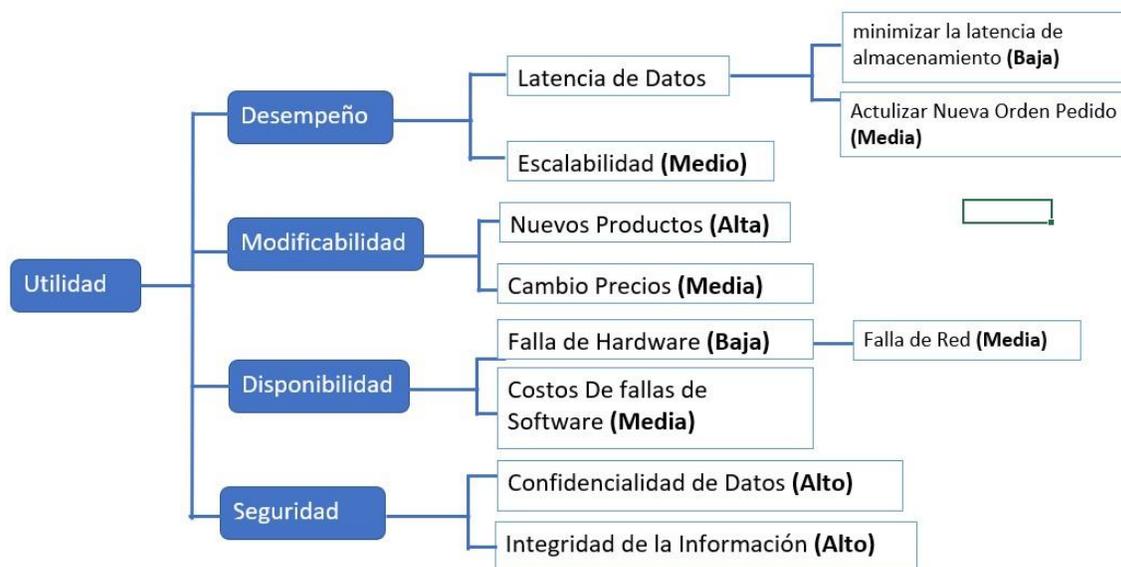


Ilustración 2. Árbol de utilidad Resumido

⁵ Wikipedia, Requisito no funcional (30-04-2021), link de consulta https://es.wikipedia.org/wiki/Requisito_no_funcional

Atributo de Calidad:	<i>Rendimiento</i>		
Latencia de Datos	ID	Descripción	Prioridad
	R001	La aplicación debe poder actualizar la orden de pedidos en el menor tiempo posible, tanto en la base de Datos en Firebase, como en cada Rol asignado en el sistema	Media, baja
Escalabilidad	ID	Descripción	Prioridad
	R002	El sistema deberá poder escalar de acuerdo con la creación de nuevos puntos de venta de la Lechonería Don Chanco	Media, Alta

Tabla 10. Rendimiento- Especificación Árbol Utilidad

Atributo de Calidad:	<i>Modificabilidad</i>		
Nuevos Productos	ID	Descripción	Prioridad
	M001	La aplicación debe poder crear o eliminar productos de acuerdo con la dinámica de la Negocio	Media, Alta
Cambio De Precios	ID	Descripción	Prioridad
	M002	El sistema debe poder realizar cambios en los precios en la lista de productos y en la orden del cliente dependiendo el nuevo precio que se le informe al cliente.	Media, Alta

Tabla 11. Modificabilidad- Especificación Árbol utilidad

Atributo de Calidad:	<i>Disponibilidad</i>		
Falla de Hardware	ID	Descripción	Prioridad

	D001	El sistema debe poder ejecutarse desde cualquier celular que cumpla con los requisitos mínimos de hardware establecidos por el desarrollador, previa autenticación del rol de cada usuario, para el caso de la caída de señal Wifi debe enviar una alerta.	Media, baja
Costos de falla de Software	ID	Descripción	Prioridad
Atributo de Calidad:	Disponibilidad		
Falla de Hardware	ID	Descripción	Prioridad
	D002	El sistema debe mitigar los posibles fallos, para evitar, posible pérdida de información	Media, Alta

Tabla 11. *Disponibilidad- Especificación Árbol Utilidad*

Atributo de Calidad:	Seguridad		
Confidencialidad de Datos	ID	Descripción	Prioridad
	S001	El sistema solo de permitir agregar Roles de usuarios, si son autorizados por el Rol administrador.	Media, Alta
Integridad de la Información	ID	Descripción	Prioridad
	S002	El sistema debe garantizar la confidencialidad de los datos de todos los clientes.	Media, Alta

Tabla 12. *Seguridad - Especificación Árbol Utilidad*

Escenario de Calidad #	1	Stakeholder: Gerencia	Prioridad: Medio
Atributo de Calidad	Escalabilidad		
Justificación	Las proyecciones de crecimiento de la empresa generan la necesidad de incluir en la estructura misma del sistema la posibilidad de soportar un número creciente de puntos de venta.		
Fuente	Usuario Final		
Estímulo	múltiples transacciones de pedidos		
Artefacto	Sistema		
Ambiente	Aplicación se encuentra en operación normal		
Respuesta	Se procesan todos los pedidos y se realizan acciones de notificación a los Stakeholders		
Medida de la Respuesta	El sistema debe procesar un promedio diario de 50 órdenes.		

Tabla 13. Escalabilidad - Escenarios de Calidad

Escenario de Calidad #	2	Stakeholder: Administrador	Prioridad: Medio
Atributo de Calidad	Nuevos Productos		
Justificación	Cambio de productos de acuerdo con la temporada		
Fuente	Usuario Final		
Estímulo	múltiples cambios en el producto		
Artefacto	Sistema		
Ambiente	Interfaz Usuario (Administrador y área de Ventas)		
Respuesta	Se procesan todos los cambios, actualizando el nuevo estado del pedido		
Medida de la Respuesta	El sistema debe procesar un promedio de 40 cambios mensuales		

Tabla 14. Nuevos Productos - Escenarios de Calidad

Escenario de Calidad #	3	Stakeholder: Administrador	Prioridad: Medio
Atributo de Calidad	Seguridad		
Justificación	La consulta y revisión de estado de pedidos requiere autenticación para que garantice que solo los usuarios autorizados tengan acceso.		
Fuente	Usuario Final		
Estímulo	El usuario debe ingresar sus credenciales para consultar información		
Artefacto	Sistema		
Ambiente	aplicación se encuentra en operación normal		
Respuesta	Sistema de verificación de identidad y control de acceso		
Medida de la Respuesta	El 100% que es usuario realice consultas		

Tabla 15. Seguridad - Escenarios de Calidad

Escenario de Calidad #	4	Stakeholder: Administrador	Prioridad: Medio
Atributo de Calidad	Seguridad		
Justificación	El sistema solo debe asignar usuarios desde el rol de Administrador		
Fuente	Administrador Punto		
Estímulo	El administrador ingresa un usuario y contraseña		
Artefacto	Sistema		
Ambiente	Aplicación se encuentra en operación normal		
Respuesta	El sistema agrega privilegios a la cuenta administrador		
Medida de la Respuesta	El 100% de los registros nuevos roles son asignados por el Administrador.		

Tabla 16. Seguridad Administrador- Escenarios de Calidad

Escenario de Calidad #	5	Stakeholder: Usuario	Prioridad: Medio
Atributo de Calidad	Seguridad		
Escenario de Calidad #	5	Stakeholder: Usuario	Prioridad: Medio
Justificación	Las características de los Usuarios de la aplicación posibilitan una fuente de amenaza que pueden significar serios problemas de seguridad para un sistema como el de gestión de pedidos como suplantación de rol.		
Fuente	Sistema individual		
Estímulo	El Usuario debe logarse.		
Artefacto	Sistema		
Ambiente	Aplicación se encuentra en operación normal		
Respuesta	Para consultar cualquier reporte el usuario debe autenticarse		
Medida de la Respuesta	El 100% de los registros nuevos roles son asignados por el Administrador.		

Tabla 17. Seguridad - Escenarios de Calidad

4.3 Diseño y arquitectura

En los siguientes diagramas muestran las diferentes vistas presentando los detalles de la arquitectura que son relevantes para cada Rol dentro de la aplicación. Mostrando los modelos que contienen información relacionada con la aplicación móvil con aspectos como su funcionamiento y la localización de los diferentes componentes en una determinada infraestructura de despliegue y la comunicación que existe con sistemas externos. Básicamente el punto de vista funcional y el de concurrencia muestran claramente que la arquitectura propuesta, se encuentra basada en arquitectura en capas, y presenta los siguiente beneficios: Escalabilidad, basadas en diferentes maquinas, el escalamiento de la aplicación es razonablemente sencillo; Flexibilidad, puede ser manejada y escalada de forma independiente; Disponibilidad, la arquitectura modular de los sistemas habilitados usado componentes que escalan fácilmente lo que incrementa la disponibilidad. Sumado a

lo anterior, se contempló la seguridad como atributo de calidad fundamental el enfoque en la autenticación de los usuarios, aplicando controles de acceso establecidos en la ISO 27001, para una posterior puesta en producción de la aplicación móvil.

A través de los modelos de esta vista podemos identificar puntos críticos del sistema sobre los cuales se deben tomar decisiones tecnológicas para favorecer el desempeño y la disponibilidad del sistema. Es importante resaltar, que se trabajan estrategias de replicación y redundancia para la correcta inclusión de la tolerancia a fallas. Se hace una inclusión mínima del punto de vista de información para mostrar el diseño propuesto del modelo ontológico, sobre el cual se edifica el modelo no relacional de bases de datos.

4.3.1 Diagrama de despliegue

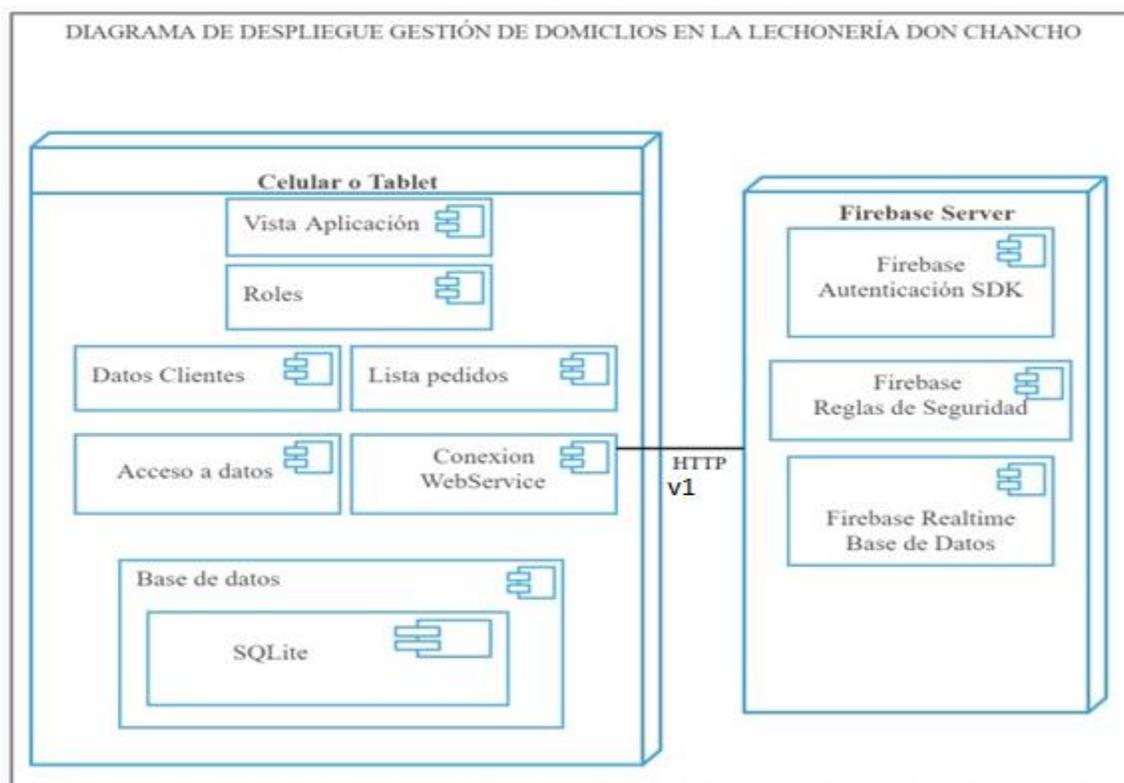


Ilustración 3. Diagrama de Despliegue

4.3.2 Caso de uso Arquitecturalmente relevante

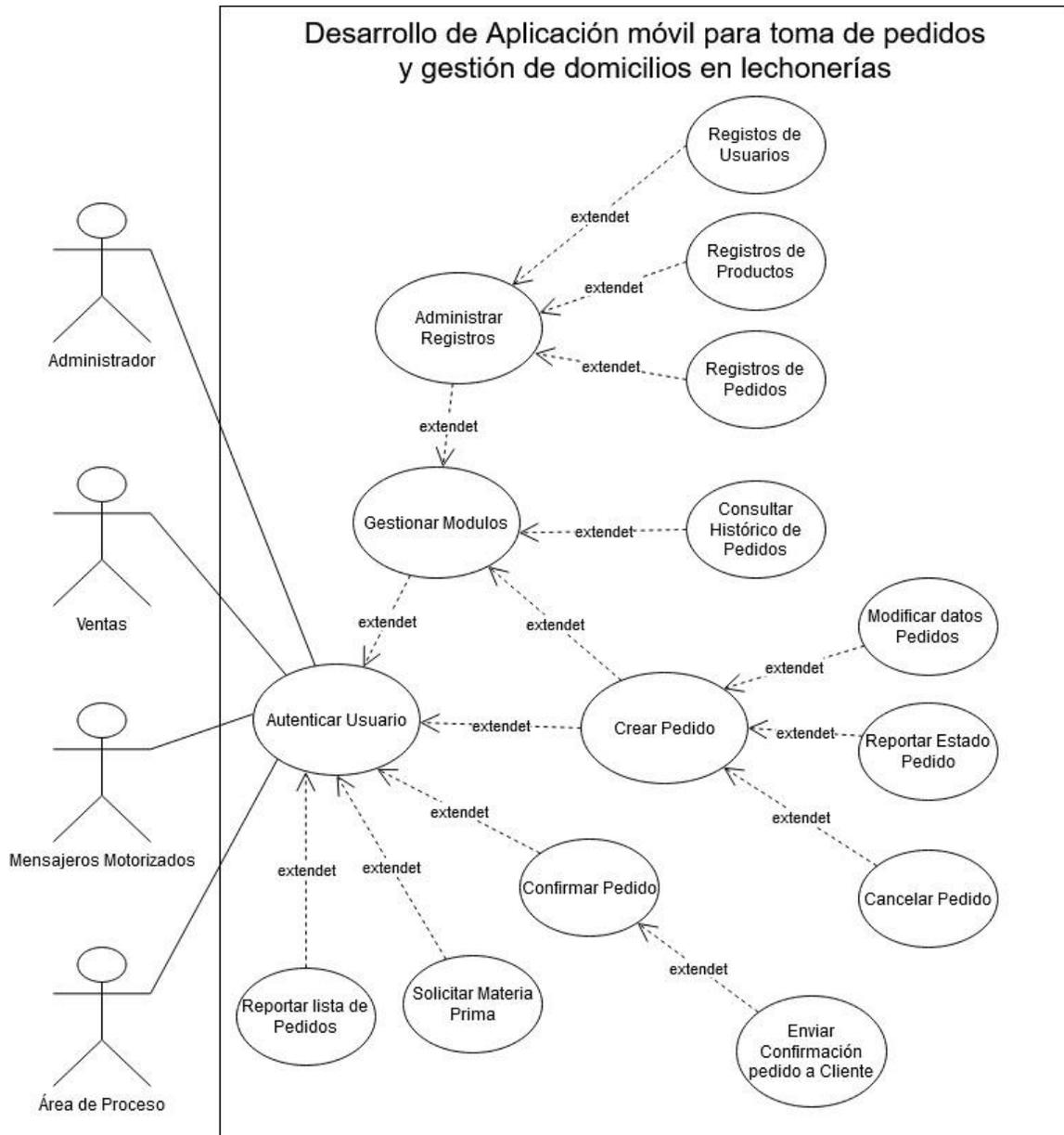


Ilustración 4. Casos de Uso

4.3.3 Diagrama de Secuencia

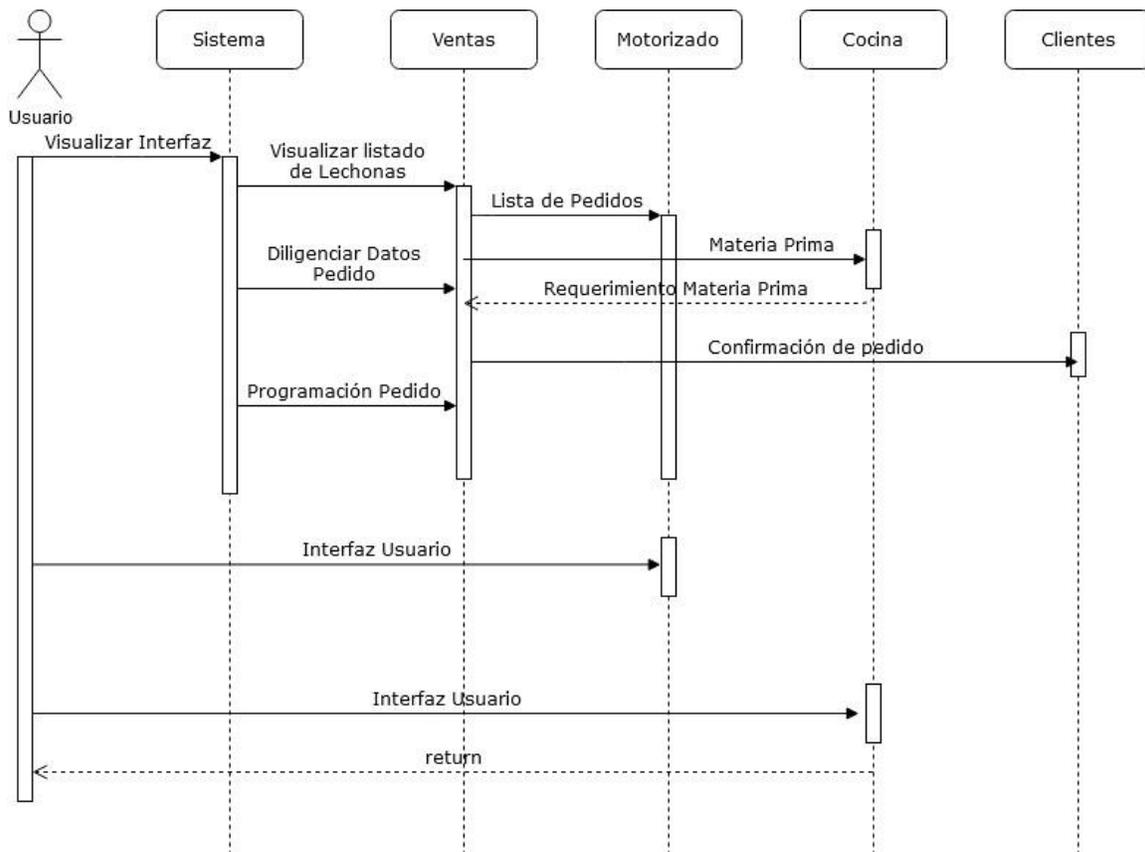


Ilustración 5. Diagrama de Secuencia

4.3.4 Diagrama de clases

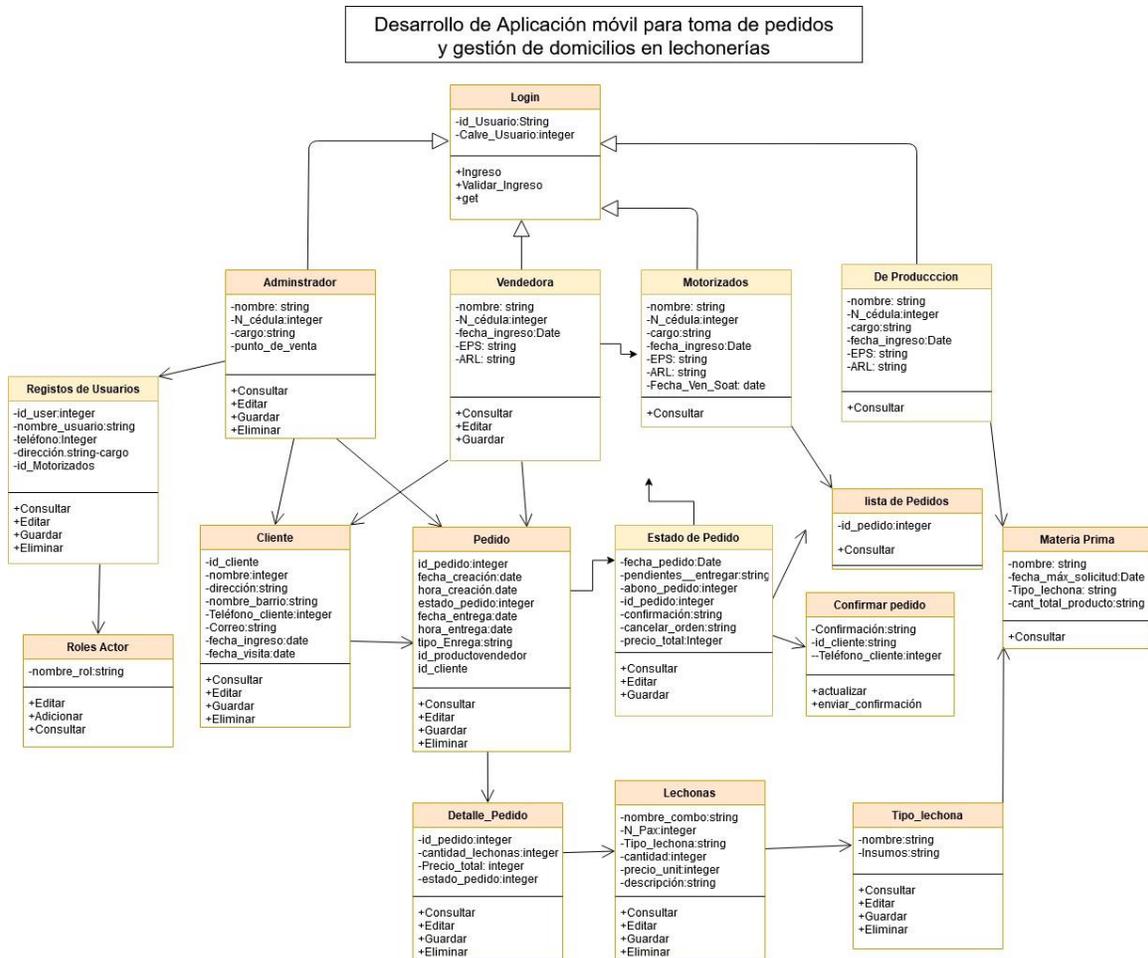


Ilustración 5. Diagrama de Clases

4.3.5 Arquitectura de alto nivel

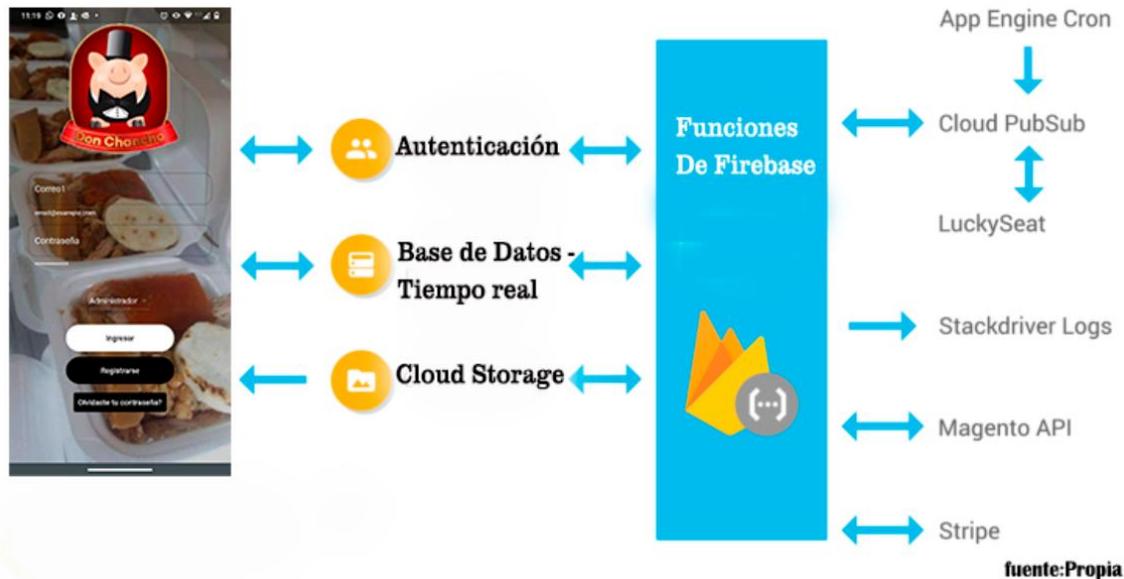


Ilustración 6. Diagrama de contexto

4.4 Construcción

En la elaboración del proyecto de desarrollo Móvil se consideró: análisis, diseño, implementación, testing y despliegue; el diseño fue realizado en adobe xd, siendo coherente con metodologías de manejo en capas de arquitectura con manteniendo una idea simple de diseño, realizando la interpretación y entendimiento década requisito del usuario final, basado en entrevistas y reuniones, así mismo después de comprender la necesidad del clientes, se implementó un diseño funcional, planteado desde un enfoque de arquitecturas en capas, debido a que esta arquitectura no es dependiente de algún framework, lo que permitió modular los componentes en forma de capas, esto facilitó la implementación y abstracción de cada una de las partes; así misio cabe destacar que esta fase se implementó con Clear Arquitectura, Patrón Bloc (bussines, Logic, Component), de esta manera se separaron los componentes de la lógica de negocios y servicios, simplemente comunicándose entre capas, en estas tenemos capas como: Data, Domain,

Device, UI, Dependency, para las pruebas se implementó de forma directa para la evaluación de componentes y lógica de la aplicación; en el despliegue, se estableció que el archivo será enviado al cliente como un archivo ejecutable que permita, la ejecución de manera sencilla en el dispositivo del usuario; a continuación se presentan algunas fotos proceso de construcción del Software.

4.4.1 Diseño de la interfaz gráfica

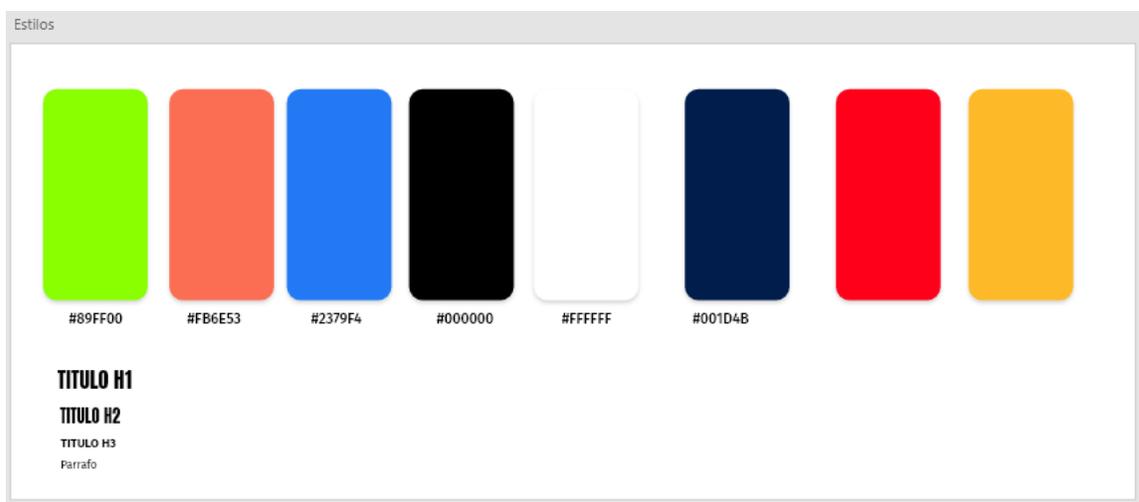


Ilustración 7. Selección de Colores y fuente

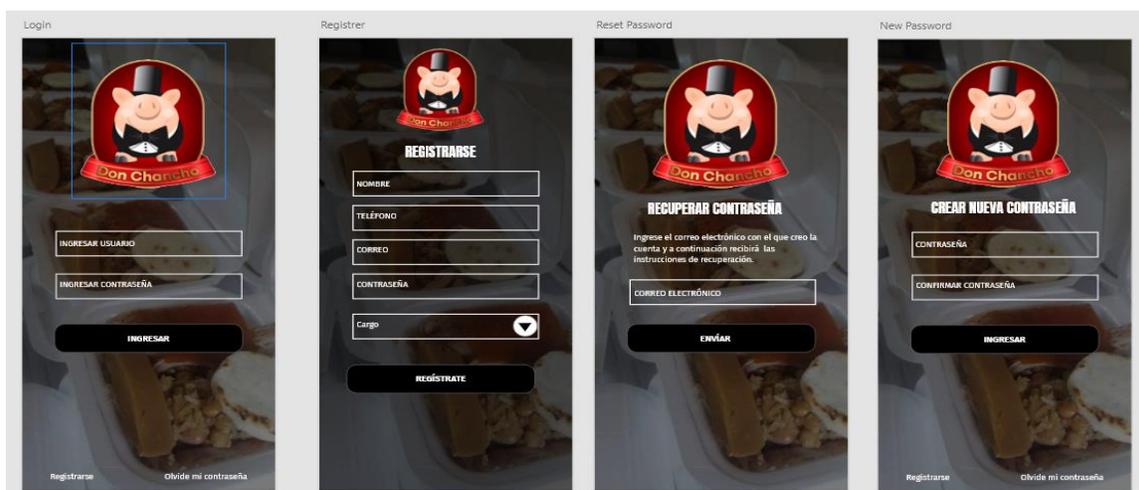


Ilustración 8. Maquetación de Inicio, Registro, recuperar contraseña, crear nueva contraseña

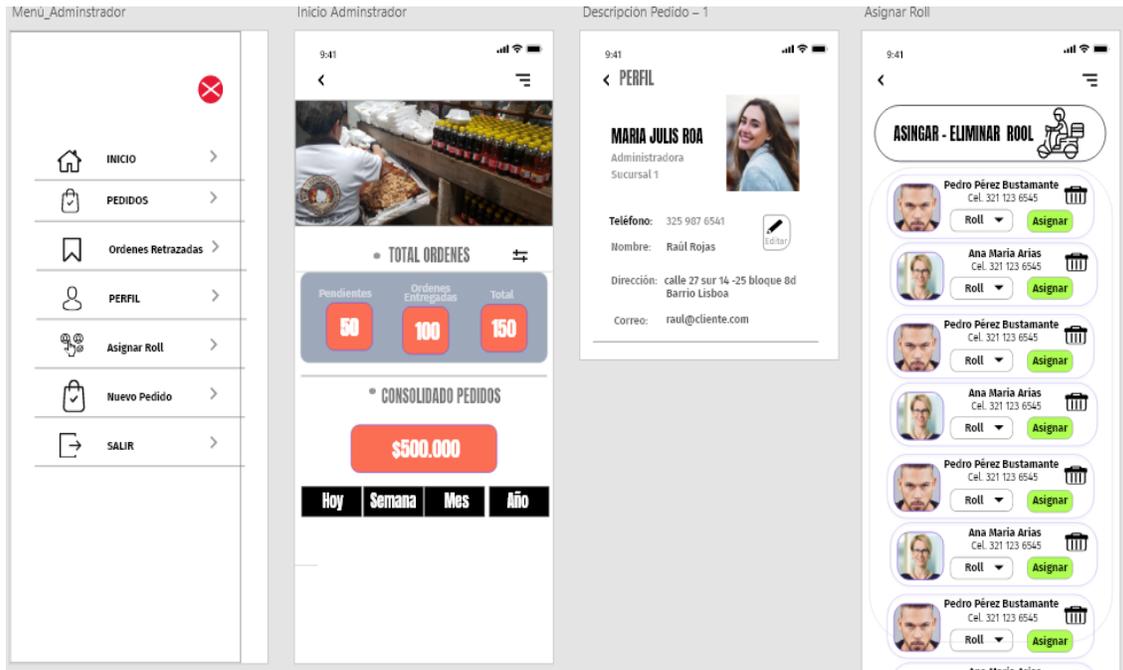


Ilustración 9. Interfaz administrador

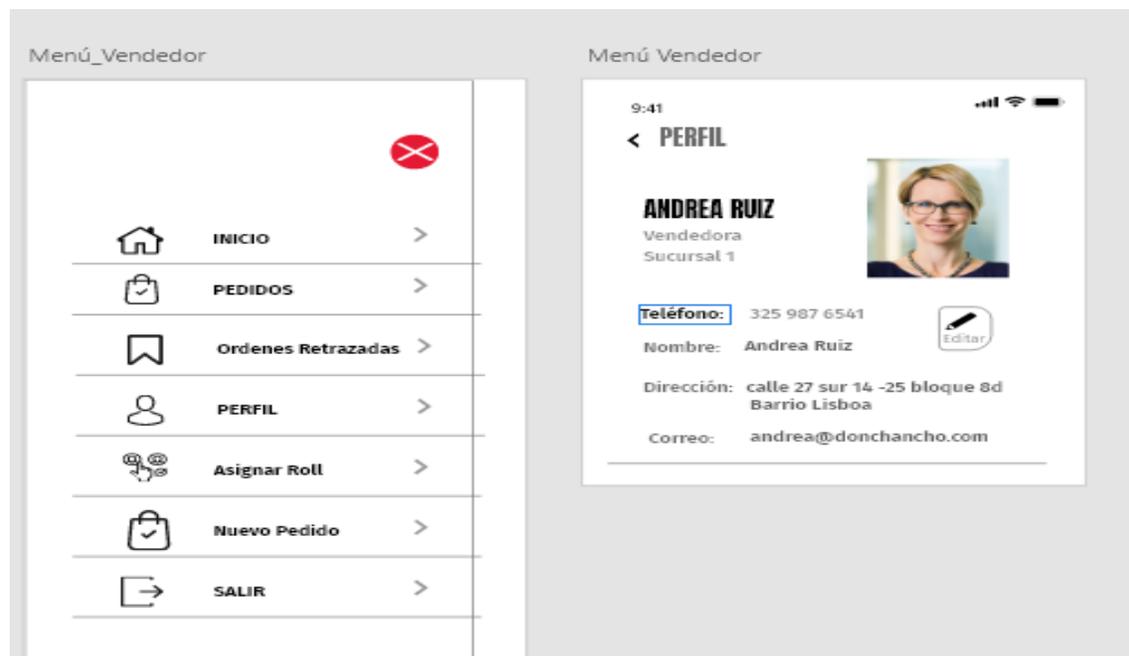


Ilustración 10. Interfaz Inicio Vendedor(a)

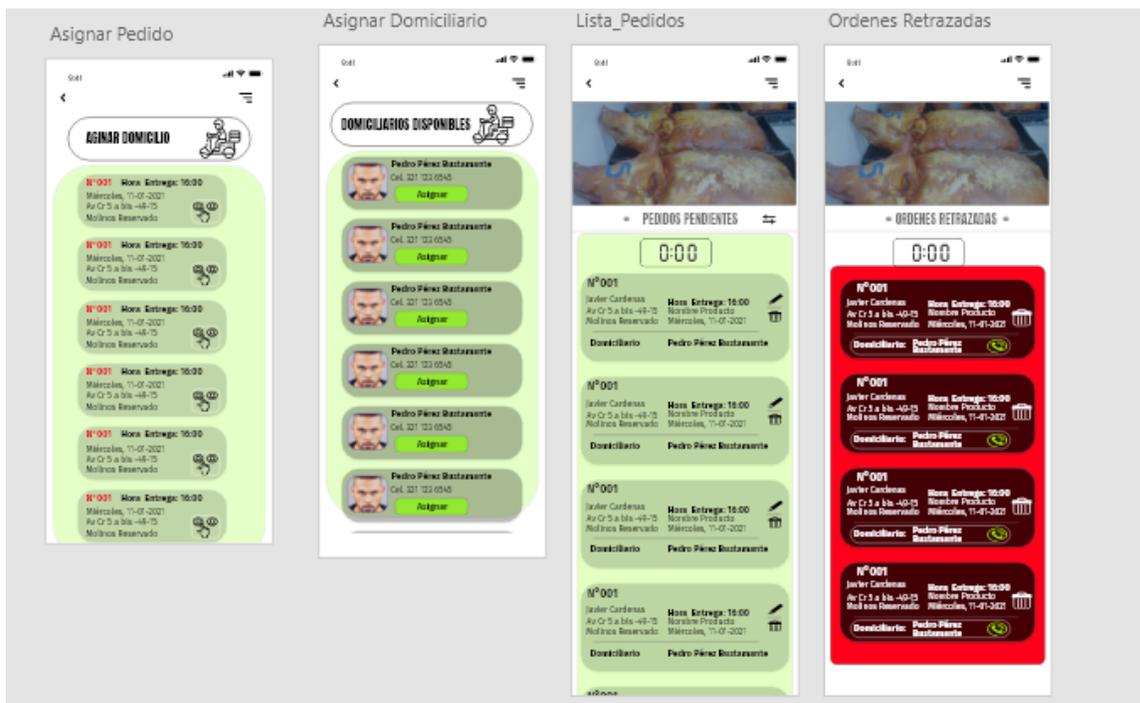


Ilustración 11. Menú común estado de pedidos - Administrador y vendedor

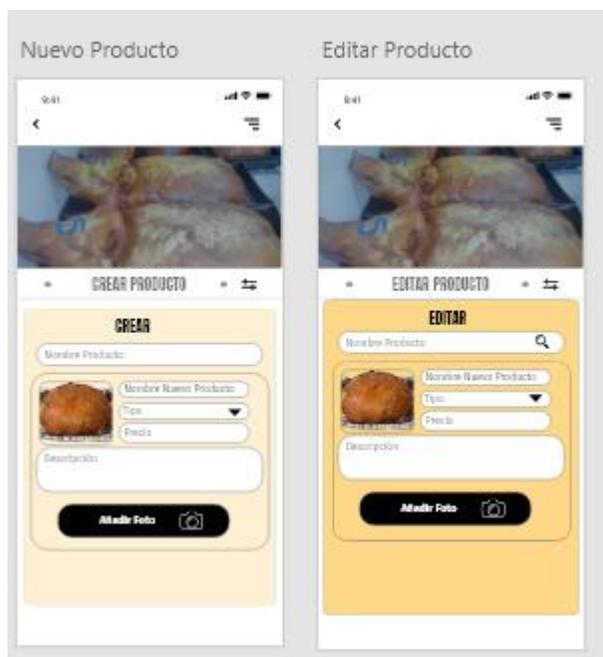


Ilustración 12. Editar producto

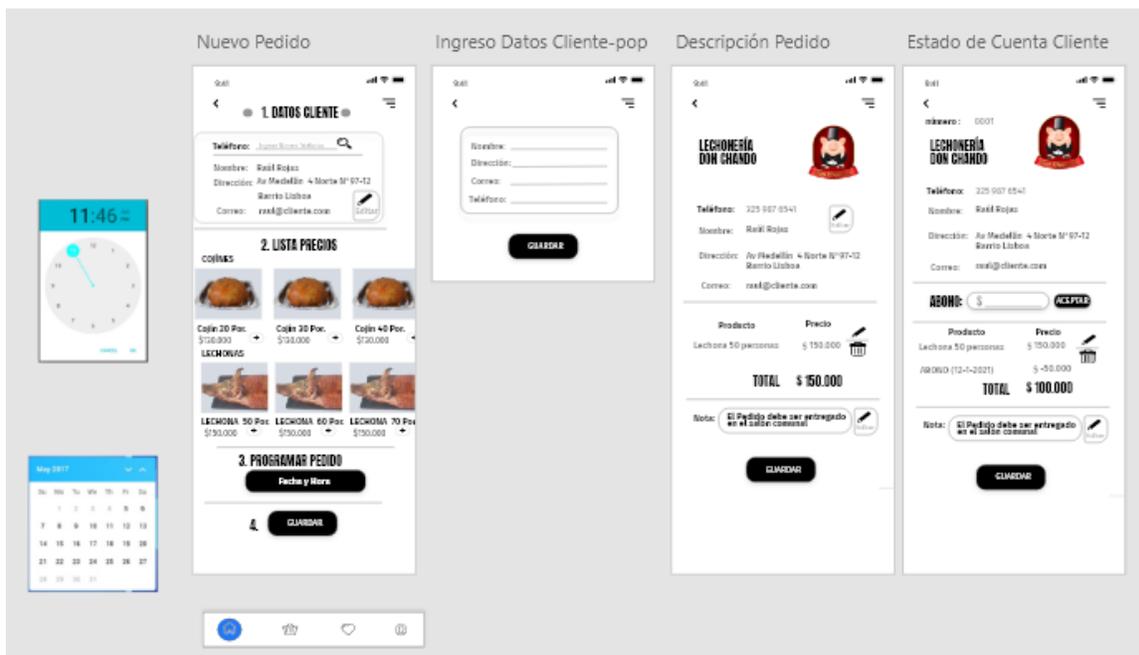


Ilustración 13. Interfaz toma de pedidos

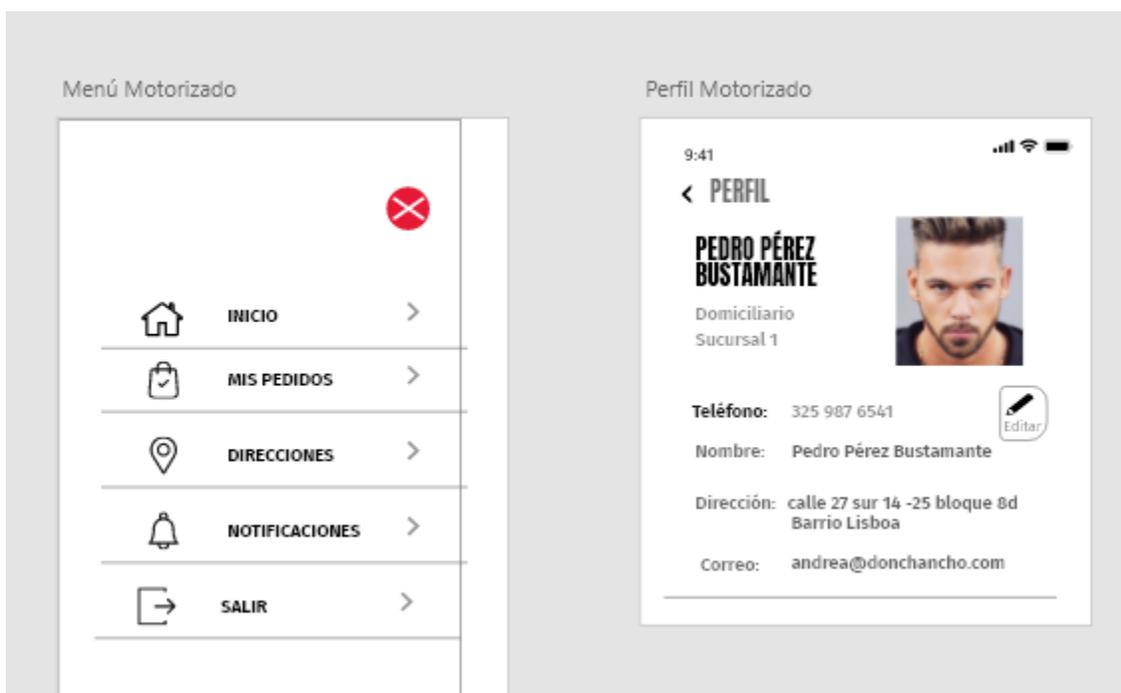


Ilustración 14. Interfaz Inicio Motorizado

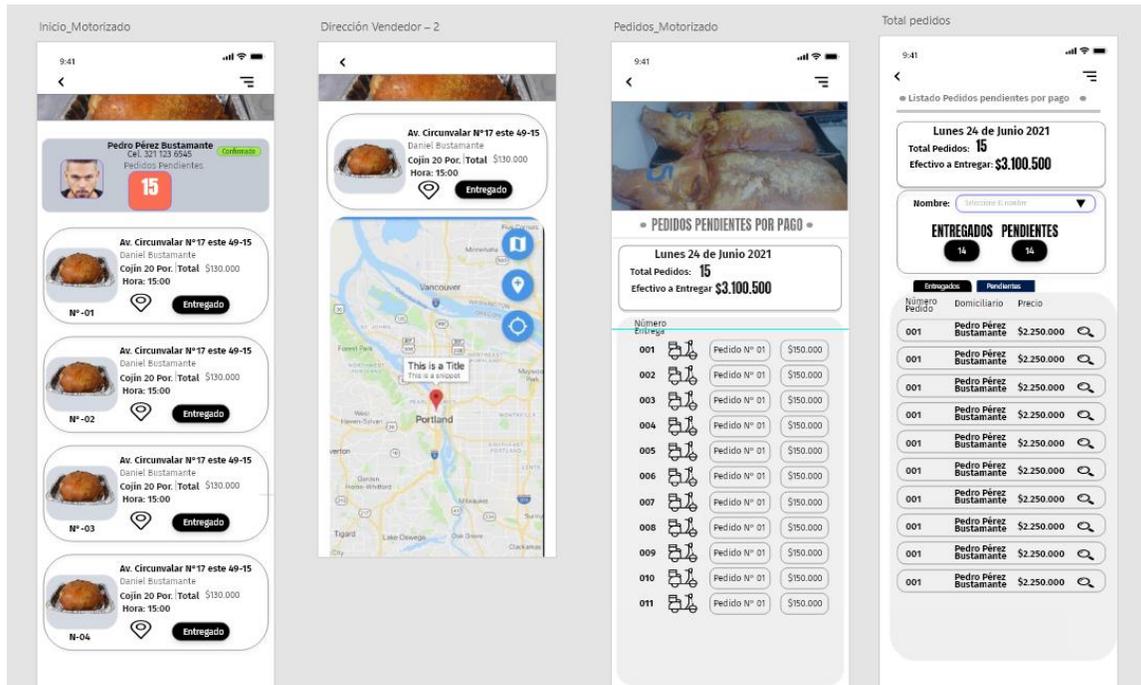


Ilustración 15. Interfaz para seguimiento de pedidos-Rol Motorizado

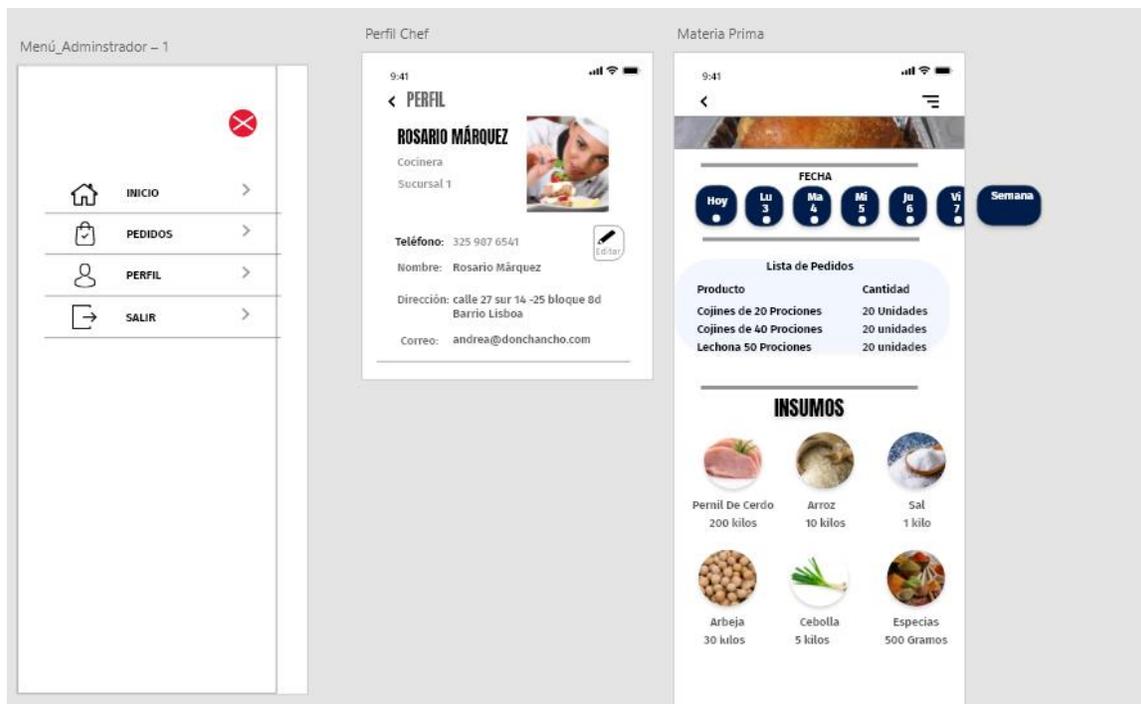


Ilustración 16. Interfaz Cocinero

4.4.2 Construcción y codificación

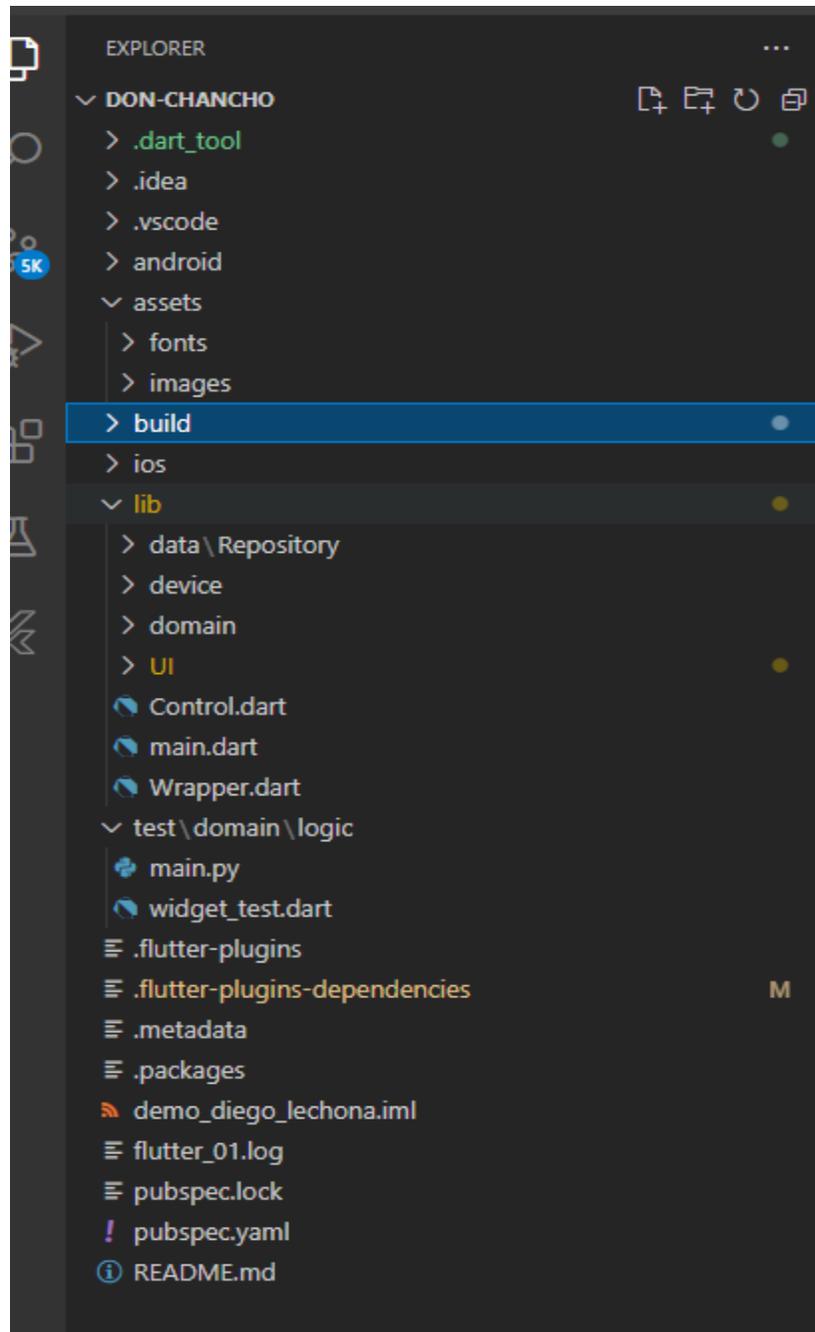


Ilustración 17. Arquitectura en Capas con el patrón – Block de Flutter

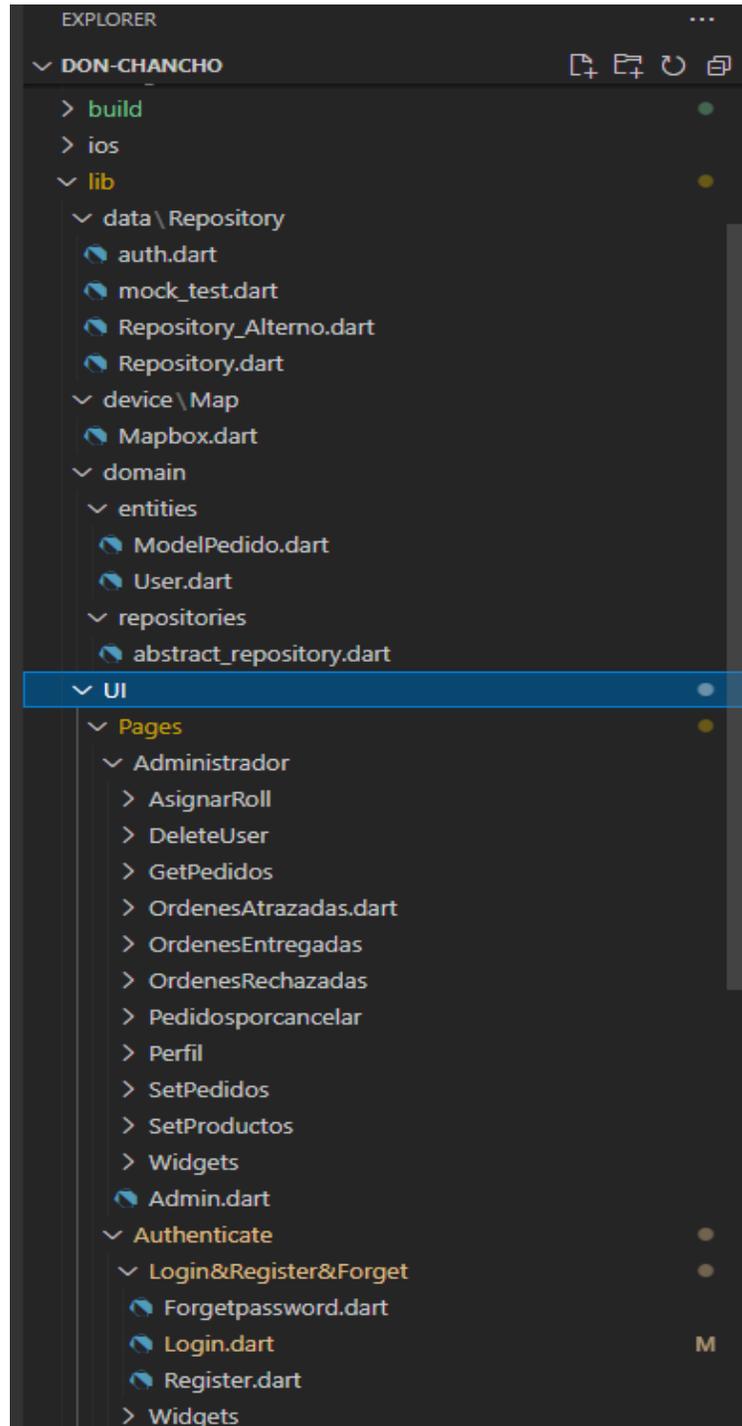


Ilustración 18. Desglose la arquitectura en capas utilizada en el proyecto

4.5 Pruebas

Para el actual proyecto una de las actividades más importantes y necesarias para garantizar la calidad de del presente desarrollo es el de realizar pruebas de calidad, con el objetivo de detectar cualquier error en la programación, garantizando que la aplicación para la gestión de domicilios cumpla con todas las características establecidas en los objetivos de presente trabajo de grado, por lo tanto una de las pruebas estipuladas para este proyecto son las pruebas funcionales, que cubra todos los casos de uso de la aplicación, que permita realizar ajustes en cuanto a la usabilidad y accesibilidad de la aplicación.

4.5.1 Prueba Inicio de Sesión

PRUEBA DE VALIDACIÓN DE INICIO DE SESIÓN	
Tipo Prueba	Funcional
Descripción	Validación Correo y Contraseña
Objetivo	Verificar los datos de ingreso del usuario para acceder a la aplicación Don Chanco, para ese caso se requerirá: correo y contraseña
Complejidad	Media
Caso N°1	
Ingresos	Correo y Contraseña
Respuestas	Acceso a la aplicación - (Ilustración 19)
Caso N°2	
Ingresos	Campos vacíos en Correo o Contraseña
Respuestas	ocurrió un error Inténtalo más tarde -(Ilustración 20)
Caso N°3	
Ingresos	Correo y Contraseña correcta y Rol Incorrecto
Respuestas	ocurrió un error Inténtalo más tarde-(Ilustración 21)

Tabla 18. Prueba funcional – Inicio de sección.

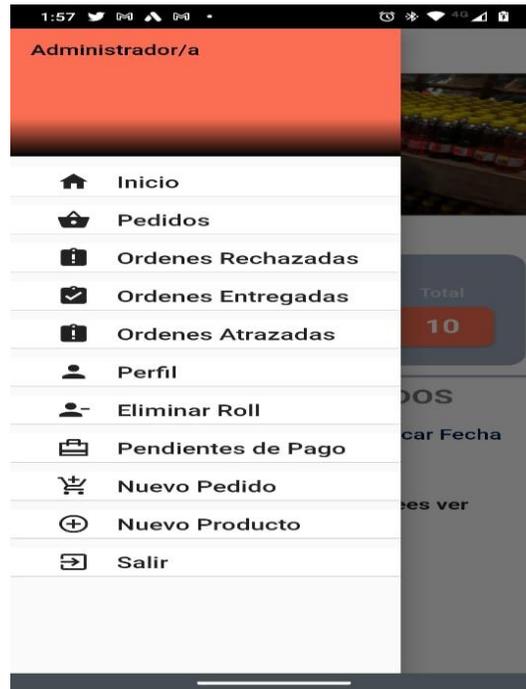


Ilustración 19. Prueba correo y Contraseña correctos, Selección Rol equivocado

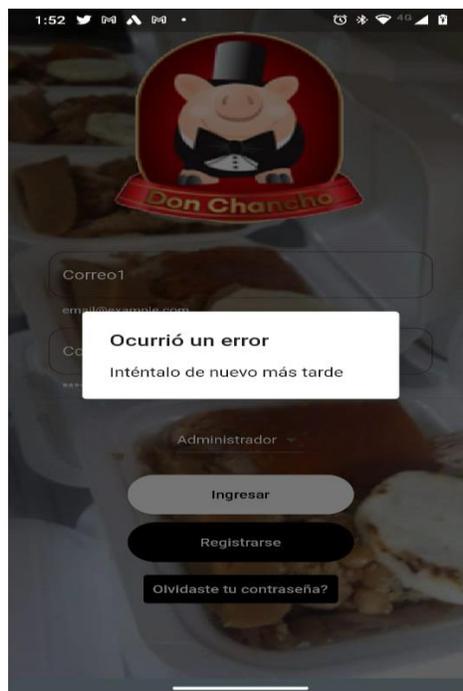


Ilustración 20. Prueba campos vacíos



Ilustración 21. Prueba correo y Contraseña correctos, Selección Rol equivocado

4.5.2 Prueba Validación Interfaz Rol Administrador

PRUEBA DE VALIDACIÓN DE INICIO DE SESIÓN	
Tipo Prueba	Funcional
Descripción	Validar la Toma de pedidos
Objetivo	Verificar el número de teléfono de un cliente ya registrado y verificar la toma de pedido en todas las fases establecidas
Complejidad	Media
Caso N°1	
Ingresos	Número de teléfono Cliente
Respuestas	Datos del cliente (Tabla 20)
Caso N°2	
Ingresos	Toma de Pedido
Respuestas	Orden de pedido (Tabla 21)
Caso N°3	
Ingresos	Seleccionar Guardar
Respuestas	El cliente debe recibir un correo de Confirmación del pedido (Tabla 22).

Tabla 19. Prueba Toma de pedidos desde el rol administrador

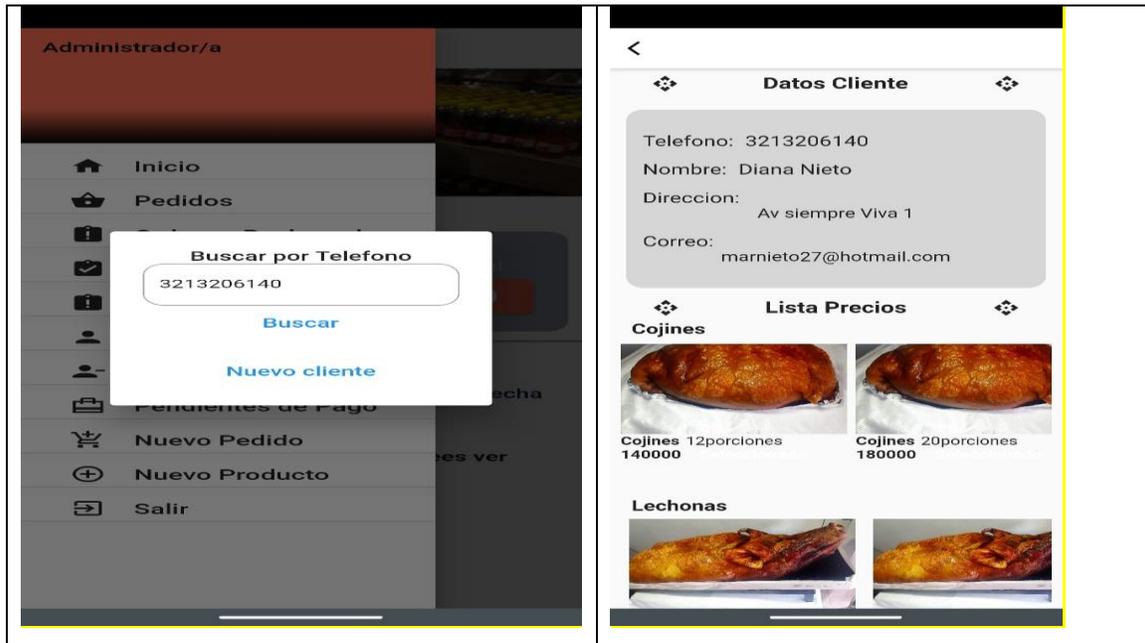
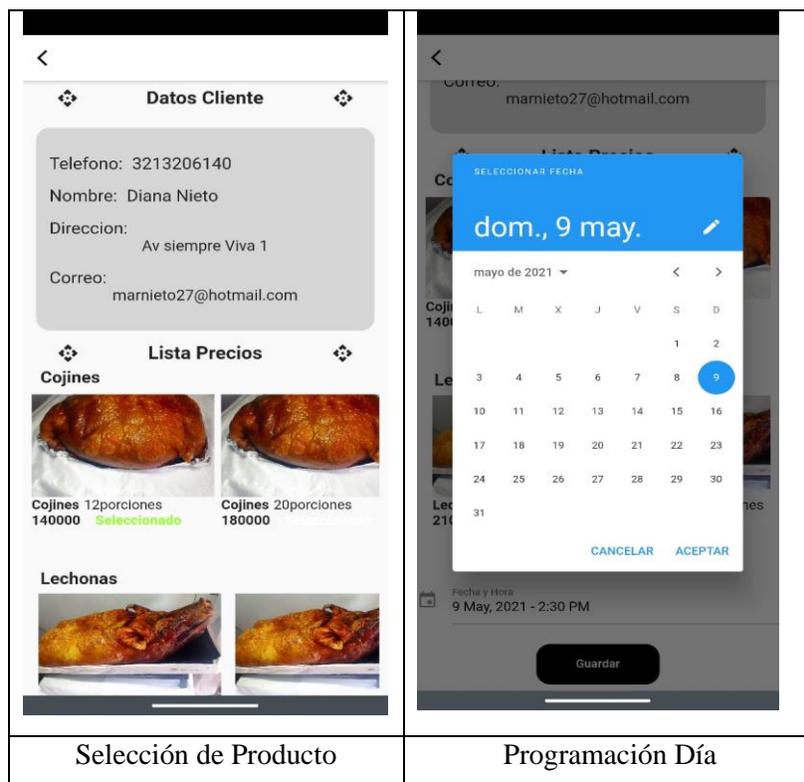
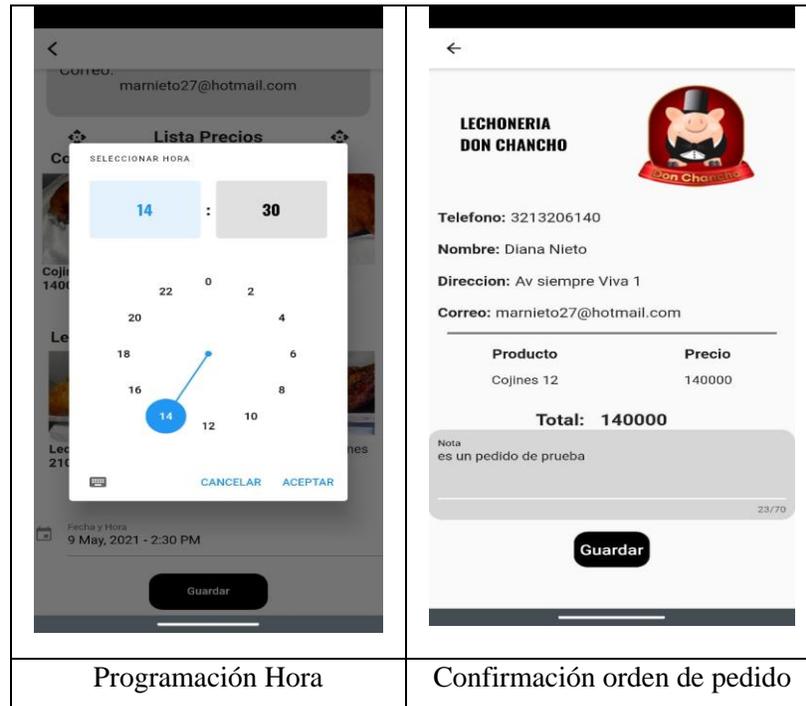


Tabla 20. Prueba Datos del cliente



Selección de Producto

Programación Día



Programación Hora

Confirmación orden de pedido

Tabla 21. Prueba toma de Pedido

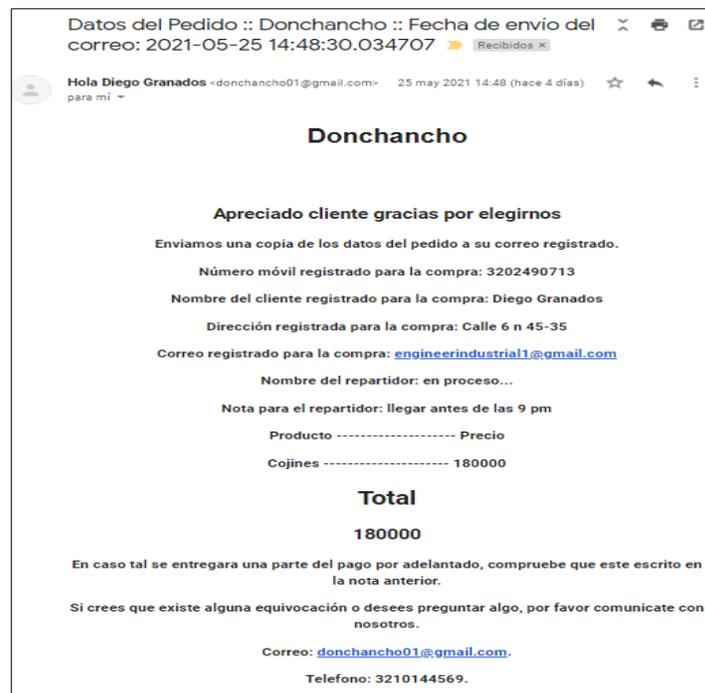


Tabla 22. Prueba Confirmación del pedido- parte 1



Tabla 23. Prueba Confirmación del pedido- parte 2

4.5.3 Prueba automatización de pruebas con BrowserStack

Se realizó la automatización de las pruebas con BrowserStack, que es una plataforma para pruebas enfocada en dispositivos móviles que permite analizar diferentes comportamientos en varios entornos con más de 100 dispositivos diferentes así mismo con varias versiones del sistema Operativo Android, sin necesidad de recurrir a dispositivos físicos o tener máquinas virtuales que funcionan de manera local consumiendo recursos; por último cabe destacar que esta herramienta tiene cuatro productos principales: App Live, Live, Automate y App Automate; donde para la actual prueba se realizó con App Automate, evidenciando lo siguiente:

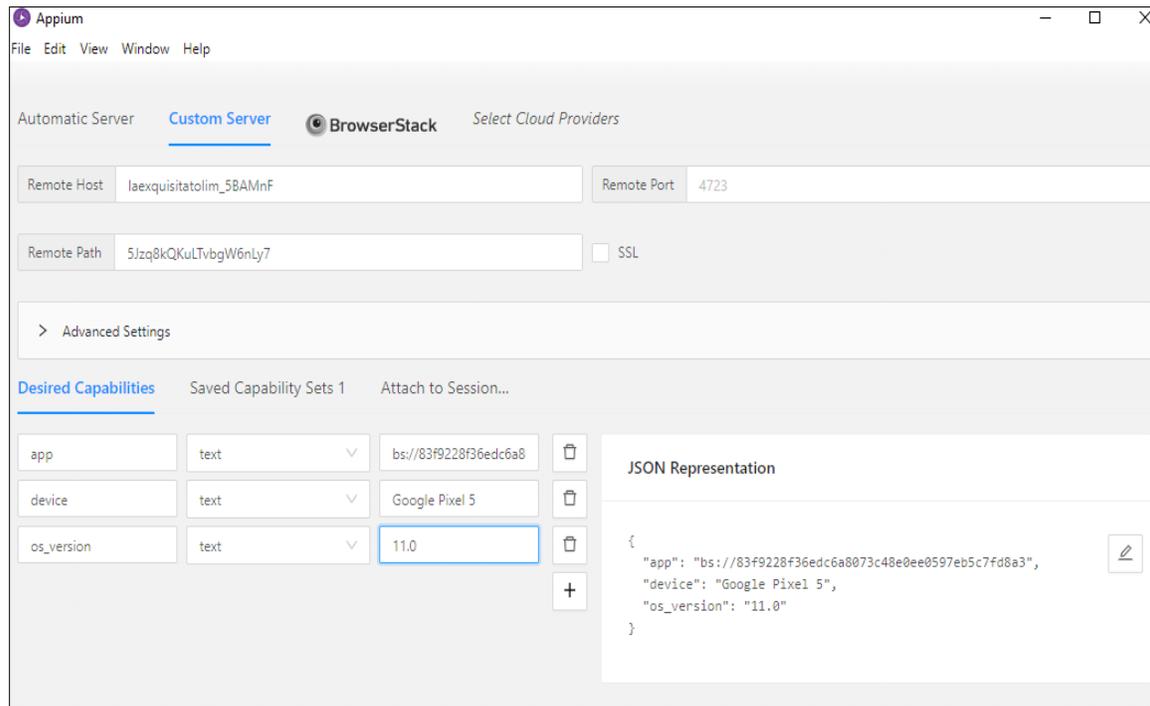


Ilustración 22. Configuración de entorno Appium

ALL SESSIONS (8) ✔ COMPLETED (2) ⚠ TIMED OUT (6) ⚪ ERROR (0) FILTER SESSIONS Search

SESSION NAME	STATUS	REST API	OS VERSION	DEVICES	DURATION	LAST UPDATED	ACTION
first_test	✔ COMPLETED	UNMARKED	ANDROID 11.0	GOOGLE PIXEL 5	23s	2 mins ago	
first_test	⚠ TIMED OUT	UNMARKED	ANDROID 11.0	GOOGLE PIXEL 5	2m 21s	14 mins ago	
first_test	⚠ TIMED OUT	UNMARKED	ANDROID 9.0	GOOGLE PIXEL 3	2m 23s	17 mins ago	
first_test	⚠ TIMED OUT	UNMARKED	ANDROID 9.0	GOOGLE PIXEL 3	2m 21s	25 mins ago	
first_test	⚠ TIMED OUT	UNMARKED	ANDROID 11.0	GOOGLE PIXEL 5	2m 20s	an hr ago	
first_test	⚠ TIMED OUT	UNMARKED	ANDROID 11.0	GOOGLE PIXEL 5	2m 23s	an hr ago	
first_test	⚠ TIMED OUT	UNMARKED	ANDROID 11.0	GOOGLE PIXEL 5	2m 24s	an hr ago	
first_test	✔ COMPLETED	UNMARKED	ANDROID 11.0	GOOGLE PIXEL 5	21s	2 hrs ago	

Ilustración 23. Pruebas realizadas en la web BrowserStack

Build > first_test
First Python project

OS: Android 9.0
Device: Google Pixel 3
Duration: 2m 28s
User: Laexquisitatolimense

Status: Error
REST API: UNMARKED
Started At: 08 May 2021 22:21 UTC
Session ID: e7505fe...

Local Testing Protocol: Off
MJSONWP

Text Logs | **Network Logs** | Device Logs | Appium Logs | App Profiling

Network Logs not enabled. To view Network Logs, set browserstack.networkLogs=true in the input capabilities for the test.

01:23 / 02:07

Ilustración 24. Prueba fallida de automatización

OS: Android 11.0
Device: Google Pixel 5
Duration: 23s
User: Laexquisitatolimense

Status: Completed
REST API: UNMARKED
Started At: 09 May 2021 04:39 UTC
Session ID: a21df985...

Local Testing Protocol: Off
MJSONWP

Text Logs | Network Logs | Device Logs | Appium Logs | **App Profiling**

Download raw data

CPU

System CPU (%) comexampdemo (%)

Memory

System Memory used (MB) System Memory available (MB) comexampdemo (MB)

Battery

Temperature (°C) Battery Charge (%)

00:15 / 00:15

25.8 101.0

Ilustración 25. Prueba de ejecución en BrowserStack

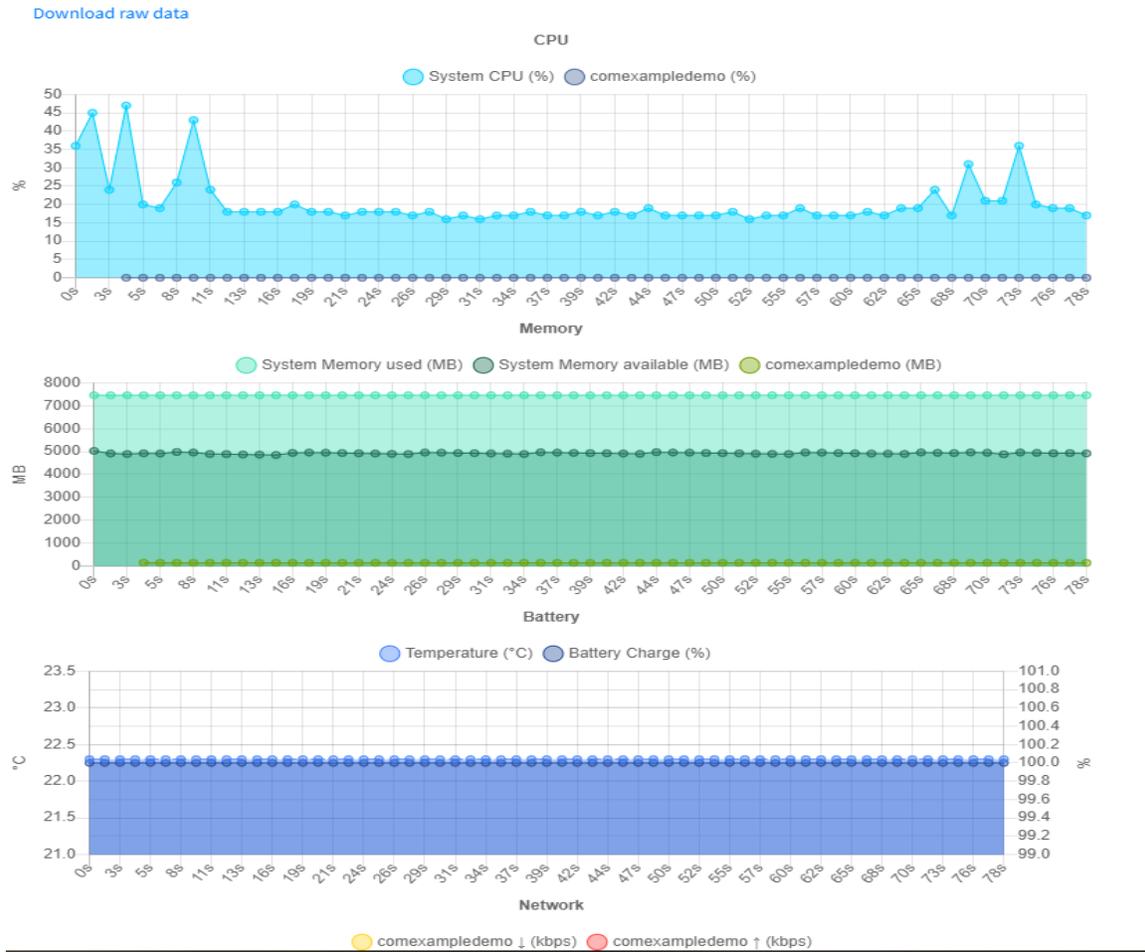


Ilustración 26. Consumo de recurso de la aplicación

4.6 Instalación y Configuración

La aplicación para la gestión de domicilios, desarrollada para Don chanco, está diseñada y optimizada para que se Instale con un archivo llamado *App-don-chancho.apk*, por la tanto se explicara cada uno de los pasos para la instalación y configuración de aplicación móvil, que fue personalizada según las necesidades de la empresa Don Chanco, donde se establecieron 4 roles principales que tienen un módulo de acuerdo con sus funciones dentro de la Lechonería, por lo tanto a continuación se explica el paso a paso para la correcta instalación y configuración.

4.6.1 Pasos para la Instalación de la Aplicación de gestión de Domicilios

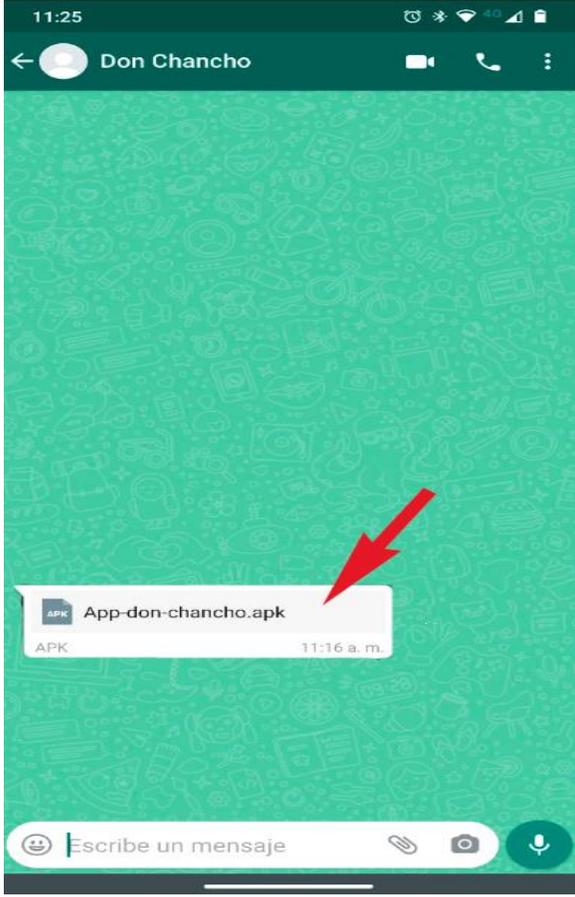
Pasos para la Instalación de la Aplicación gestión de Domicilios	
	
<p>Nota: El archivo ejecutable App-don-chancho.apk, será enviado a través de WhatsApp.</p> <p>Primer paso: Abrir la aplicación tocando la pantalla dos veces seguidas sobre al archivo llamado "App-don-chancho.apk "</p>	<p>Segundo Paso: Seleccionamos la opción abrir</p> <p>Nota: La aplicación es de origen confiable y no va a dañar su sistema</p>

Tabla 24. Instalación y Configuración primer y segundo paso

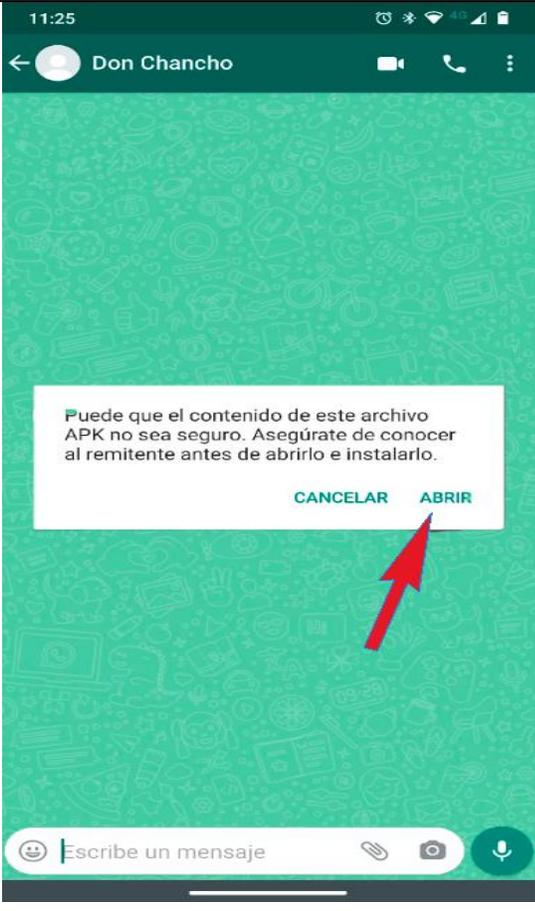
Pasos para la Instalación de la Aplicación gestión de Domicilios	
	
<p>Tercer paso: Seleccionar configuración</p>	<p>Cuarto Paso: Seleccionamos la opción abrir Nota: La aplicación es de origen confiable y no va a dañar su sistema</p>

Tabla 25. Instalación y Configuración tercer y cuarto paso

Pasos para la Instalación de la Aplicación gestión de Domicilios	
 <p style="text-align: center;">WhatsApp 2.21.8.17</p> <p>Confiar en esta fuente <input type="checkbox"/></p> <p>Tu teléfono y tus datos personales son más vulnerables a los ataques de apps desconocidas. Si instalas apps de esta fuente, aceptas tu responsabilidad respecto de los daños que sufra tu teléfono o la pérdida de datos que pueda producirse a partir de su uso.</p>	 <p style="text-align: center;">WhatsApp 2.21.8.17</p> <p>Confiar en esta fuente <input checked="" type="checkbox"/></p> <p>Tu teléfono y tus datos personales son más vulnerables a los ataques de apps desconocidas. Si instalas apps de esta fuente, aceptas tu responsabilidad respecto de los daños que sufra tu teléfono o la pérdida de datos que pueda producirse a partir de su uso.</p>
Quinto paso: Activar “Confiar en esta fuente”	

Tabla 26. Instalación y Configuración quinto paso

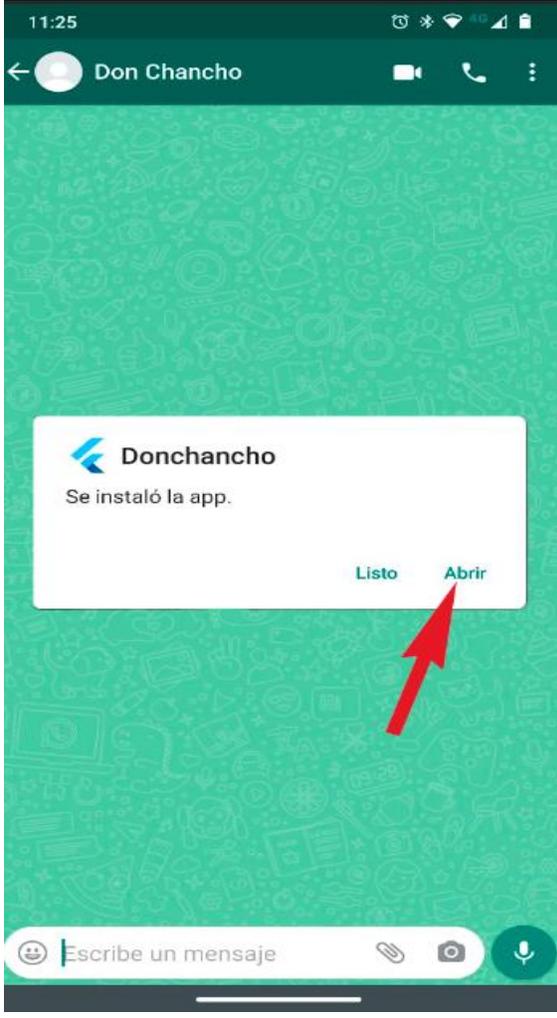
Pasos para la Instalación de la Aplicación gestión de Domicilios	
 <p>11:25</p> <p>Don Chancho</p> <p>Donchancho</p> <p>¿Deseas instalar esta aplicación?</p> <p>Cancelar Instalar</p> <p>Escribe un mensaje</p>	 <p>11:25</p> <p>Don Chancho</p> <p>Donchancho</p> <p>Se instaló la app.</p> <p>Listo Abrir</p> <p>Escribe un mensaje</p>
Sexto paso: Seleccionar Instalar.	Séptimo paso: Seleccionar Abrir.

Tabla 27. Instalación y Configuración sexto y séptimo paso

Pasos para la Instalación de la Aplicación gestión de Domicilios



Octavo paso: Para la última parte de la configuración e instalación, aparecerá la pantalla de inicio de aplicación en el cual debemos registrarnos, seleccionando el rol que tengamos asignado, dentro de la Lechonería

Tabla 28. Instalación y Configuración Octavo Paso

4.6.1.1 Destilación de la Aplicación

Si por alguna razón desea desinstalar la aplicación de su dispositivo móvil, siga estos pasos:

- Abra el menú de aplicaciones
- Busque la aplicación Don Chanco para desinstalar
- Presione y aguante el icono de la aplicación, hasta que la pantalla cambie y aparezcan nuevas opciones.
- Continúe presionando el icono.
- Arrastre el icono hacia “Desinstalar”
- Confirme que desea remover

4.7 Conclusiones

- En el anterior en el trabajo grado he consigo adquirir habilidades fuertes y débiles, aplicando lo visto en las diferentes asignaturas de la especialización en ingeniería de Software, así mismo se ha profundizado en los conocimientos en el Framework Flutter; tanto en el diseño de la interfaz gráfica como en el desarrollo basado en Clean Arquitectura, aplicado para aplicaciones móviles basados con el sistema operativo Android, así mismo me ha permitido aprender Flutter, un lenguaje con el que no estaba familiarizado y que es ampliamente utilizado para crear aplicaciones móviles con una interfaz amigable, mejorando la experiencia del usuario final, lo anterior implementando una arquitectura en capas. Además, me ha posibilitado gestionar las diferentes tareas mediante la metodología Ágil Kanban y reforzar mis conocimientos sobre la herramienta; en el ámbito profesional ha sido mi primer contacto formal con la Arquitectura de Software, aplicándola para la gestión de domicilios en lechonerías.
- Se ha cumplido el objetivo y requisitos mencionados en el presente trabajo de grado, donde se pretendía construir una solución móvil para la gestión de domicilios, que fuera capaz de realizar el seguimiento al estado del pedido, así mejorando la comunicación en los diferentes roles que se establecieron en el presente desarrollo, todo lo anterior permitiendo él envió a través de correo electrónico de la confirmación del pedido al cliente.
- Una de las lecciones aprendidas más importantes, es la importancia de aplicar una buena arquitectura para cualquier desarrollo de software, sin importar el tamaño, así como aplicar correctamente las fases de: Planificación, Análisis, Diseño, Implementación y Pruebas; que resultaron ser de suma importancia para el éxito del presente desarrollo móvil basado en el SDK Flutter.

5 Referencias

- Ximena Muñoz, (22-03-2021). Lechonería Charrasco Link de referencia <https://sites.google.com/site/lechoneriacharrasco/noticias/lahistoriadelalechona>
- Andrés A. Gómez Martín. (22-03-2021). La lechona: una historia deliciosa, Link de referencia: <https://www.elcampesino.co/la-lechona-una-historia-deliciosa/>
- Modelado sistemas utilizando Puntos de Vista (Viewpoint-Based Modeling), (22-03-2021), link de referencia, <https://eaminds.com/2017/07/28/disenando-software-en-varias-perspectivasviewpoint-modeling/>
- Diagrama de despliegue, tomado de <https://diagramasuml.com/despliegue/>
- Documento de arquitectura (SAD) (22-03-2021), link de referencia https://profesores.virtual.uniandes.edu.co/~isis2503/dokuwiki/lib/exe/fetch.php?media=proyectos:casos_estudio:hurtadoramosusa201120.pdf
- Implementación de interfaz de modelado, (22-03-2021), link de referencia http://oa.upm.es/35520/1/TD_PEREZ_COSTOYA_FERNANDO.pdf
- Software Architecture Document (22-03-2021), link de referencia [https://cmapspublic2.ihmc.us/rid=1NPZT4HTD-1N402VM-22L8/sad-up%20\(\).pdf](https://cmapspublic2.ihmc.us/rid=1NPZT4HTD-1N402VM-22L8/sad-up%20().pdf)
- Trifeniz, Arquitectura Sin Servidores y Microservicios, Consultado (30-04-2021), link de referencia <https://www.trifenix.io/arquitectura-sin-servidores-y-microservicios/>